ABSTRACT

Title of Dissertation: COMBINATORIAL OPTIMIZATION ALGORITHMS FOR HYPERGRAPH MATCHING WITH APPLICATION TO POSTURE IDENTIFICATION IN EMBRYONIC *CAENORHABDITIS ELEGANS*

Andrew Neil Lauziere
Doctor of Philosophy, 2022

Dissertation Directed by: Professor Radu Balan
University of Maryland Center for Scientific Computing & Applied Mathematical Modeling

Point-set matching defines the task in computer vision of identifying a one-to-one alignment between two sets of points. Existing techniques rely on simple relationships between point-sets in order to efficiently find optimal correspondences between larger sets. Modern methodology precludes application to more challenging point-set matching tasks which benefit from interdependent modeling. This thesis addresses a gap in combinatorial optimization literature by enhancing leading methods in both graph matching and multiple object tracking (MOT) to more flexibly use data-driven point-set matching models. Presented contributions are inspired by *Caenorhabditis elegans*, a transparent free-living roundworm frequently studied in developmental biology and neurobiology. The *C. elegans* embryo, containing around 550 cells at hatch, can be used for cell tracking studies to understand how

cell movement drives the development of specific embryonic tissues and organ functions. The development of muscle cells complicates analyses during late-stage development, as embryos begin twitching due to muscular activity. The sporadic twitches cause cells to move violently and unpredictably, invalidating traditional cell tracking approaches. The embryo possesses *seam cells*, a set of 20 cells which together act as fiducial markers to approximate the coiled embryo's body. Novel optimization algorithms and data-driven hypergraphical models leveraging the correlated movement among seam cells are used to forward research on *C. elegans* embryogenesis. We contribute two optimization algorithms applicable in differing conditions to interdependent point-set matching. The first algorithm, *Exact Hypergraph Matching* (EHGM), exactly solves the n-adic assignment problem by casting the problem as *hypergraph matching*. The algorithm obtains solutions to highly interdependent seam cell identification models. The second optimization algorithm, *Multiple Hypothesis Hypergraph Tracking* (*MHHT*), adapts traditional multiple hypothesis tracking with hypergraphical data association. Results from both studies highlight improved performance over established methods while providing adaptable optimization tools for multiple academic communities.

The canonical point-set matching task is solved efficiently under strict assumptions of frame-to-frame transformations. Challenging situations arising from non-rigid displacements between frames will complicate established methods. Particularly, limitations in fluorescence microscopy paired with muscular twitching in late-stage embryonic *C. elegans* yield adversarial point-set matching tasks. Seam cell identification is cast as an assignment problem; detected cells in images are uniquely identified through a combinatorial optimization algorithm. Existing graph matching methods are underequipped to contextualize the coiled embryonic position in sparsely imaged samples. Both the lack of an effective point-set matching

model and an efficient algorithm for solving the resulting optimization problem limit computationally driven solutions to identify seam cells in acquired image volumes. We cast the n-adic problem as hypergraph matching and present an exact algorithm to solve the resulting optimization problem. EHGM adapts the branch-and-bound paradigm to dynamically identify a globally optimal correspondence; it is the first algorithm capable of solving the underlying optimization problem. Our algorithm and accompanying data-driven hypergraphical models identify seam cells more accurately than established point-set matching methods.

The final hours of embryogenesis encompass the moments in which *C. elegans* assumes motor control and begins exhibiting behavior. Rapid imaging of the seam cells provides insight into the embryo's movement as a proxy for behavior. However, seam cell tracking is especially challenging due to both muscular twitching and the low dose required to gently image the embryo without perturbing development. Current methods in MOT rely on independent object trajectories undergoing smooth motion to effectively track large numbers of objects. Multiple Hypothesis Tracking (MHT) is the foremost method for challenging MOT tasks, yet the method cannot model correlated object movements. We contribute Multiple Hypothesis Hypergraph Tracking (MHHT) as an extension of MHT, which performs interdependent object tracking by jointly representing objects as a hypergraph. We apply MHHT to track seam cell nuclei during late-stage embryogenesis. Data-driven hypergraphical models more accurately track seam cells than traditional MHT based approaches. Analysis of time-lapse embryonic postures and behavioral motifs reveal a stereotyped developmental progression in *C. elegans*. Further analysis uncovers late-stage motility defects in *unc-13* mutants.

COMBINATORIAL OPTIMIZATION ALGORITHMS FOR HYPERGRAPH MATCHING WITH APPLICATION TO POSTURE IDENTIFICATION IN EMBRYONIC *CAENORHABDITIS ELEGANS*

by

Andrew Neil Lauziere

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2022

Advisory Committee:
Professor Radu Balan, Chair/Advisor
Dr. Hari Shroff, Co-Advisor
Associate Professor Maria Cameron
Associate Professor Vincent Lyzinski
Professor Eric Haag

Preface

Multidisciplinary research aims to unite faculty from different disciplines to approach a shared academic challenge. Applied mathematics stands at the intersection of various disciplines to bind experts together towards a shared goal. For example, statisticians are integral across domains in the natural and social sciences to model signal and error arising in samples. Moreover, computer scientists are becoming increasingly ubiquitous to design computational methods and software not only to realize theoretical mathematical solutions but to market research to their respective academic communities. Practitioners in such roles stand to unite disparate academic communities in pursuit of biological discovery.

Research comprising this thesis was pursued from the perspective of a mathematician bringing together biologists and optical physicists. Each discipline served a role in the partnership. Biologists hypothesized within their niche as to the nature of the world. The optical physicist built microscopes in which to observe the biological specimen. The mathematician then developed computational tools to analyze and model the observed biological data. Statistical models processed the data to yield new biological insight.

In particular, the biologists studied embryo-development of the nematode *Caenorhabditis elegans*. The roundworm has reached such ubiquity across scientific domains that it is used as a model organism for development. The relative simplicity of the worm makes it an ideal candidate for understanding development. However, this simplicity comes with a tradeoff. Observing the embryo as it develops is uniquely challenging due to its fragility

and movement late in development. The biological questions of interest center about the final stages of embryo-development in which observation becomes especially adversarial due to twitching motions caused by muscle cells. Microscopists have developed cutting edge microscopes and accompanying software just capable of observing the embryo in these last hours prior to hatching. These cutting edge tools enable observation of an otherwise unstudied phenomena. The resulting image data itself presents new concerns as they are not fully amenable to existing methods of analysis. My role in these interdisciplinary projects was to develop mathematical tools to analyze images of *C. elegans* embryos to extract information relevant to biological research questions. Contributions from each of the three disciplines helped yield new biological insight concerning *C. elegans* tissue formation and behavioral development.

Acknowledgments

Support for my journey has spanned from individuals to institutions. Two institutions and their union made this possible: University of Maryland, College Park and National Institutes of Health. The University of Maryland has been my home for the past decade; staff and faculty across campus are responsible for not just my education but for creating opportunities for me to grow and contribute. On the other hand, the National Institutes of Health provided opportunities to research and collaborate. Both institutions gave me opportunities to teach, mentor, and enrich each community, respectively.

The staff at UMD, from Resident Life, to Dining Services, Recreation & Wellness, the University Health Center, and beyond comprise the foundation of UMD. While educators across Behavioral and Social Science, Computer, Mathematical, and Natural Sciences provided their time and attention for me to learn, the staff across departments made it such that the learning was possible.

Then, the National Institutes of Health, particularly the National Institute of Biomedical Imaging and Bioengineering, helped give me the tools to be a researcher. Opportunities to collaborate, present, and meet other scientists gave insight into performing scientific research.

While these institutions have done so much to support me, they are ultimately comprised of individuals. Foremost, I'd like to thank my committee: Radu Balan, Hari

Shroff, Maria Cameron, Vince Lyzinski, and Eric Haag. My advisors Radu and Hari have each worked tirelessly to support me. Within the Mathematics department, Haydee Hidalgo, Jessica Sadler, Ruth Yun, Kate Truman, and Liz Wincek have been integral as advisors, friends, and staff to my success. Faculty members Ben Kedem, Denny Gulick inspired me and pushed me over many years in our department. Outside of Mathematics, Carrie Jacobs of the Health Center has been a great friend, medical provider, and advisor throughout my doctoral program.

Lab members within High Resolution Optical Imaging provided a second home for me. Collaborators Ryan Christensen, Evan Ardiel, and Stephen Xu supported my progress and ensured I was always correctly describing *C. elegans*. Amy Elliot, Yijun Su, and Nensi Karaj supported me through conversations and friendship. I again thank Hari not just for opening his lab to me, but for being a friend.

Friends and family have done so much over the past ten years. My parents, step parents, and brother have together always made sure I am supported. Friends I've made along the way: Mark Wymer, Jordan Hirsh, Kiante Brantley, Denis Peskov, Kyle Sheppard, Casey Cavanagh, Maddy Gray, Shamar Rundawaha, Ahmed Abdosh, and Audrey Paranka have become a second family. Friends that have been with me: Dan Hill, Dylan Thompson, Michael Soto, David Wood, Andrew Pasqaulle, and Kristen Miller have seen every up and down over the years, but have always been by my side. We did this together.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Overview

Advancements in the natural sciences are becoming increasingly predicated on collaboration from experts spanning multiple domains. Biologists in particular are benefiting from the partnership with mathematicians and computer scientists. Data comprised of high resolution images spanning multiple hours must be analyzed to extract relevant biological information. Computational methods operating on image data have been shown to accelerate analyses, enabling biological researchers to perform wider arrays of experiments, enabling the discovery of richer biological phenomena [78].

Presented contributions are motivated by the nematode *Caenorhabditis elegans* (*C. elegans*). The roundworm is studied as a model of developmental biology due to myriad facets of the organism; Sydney Brenner, the original investigator of the roundworm, called *C. elegeans* "Nature's gift to science" in his award speech for the Nobel Prize in Medicine [12]. The following two sections each present foundational information concerning *C. elegans* biology, and foundations of microscopy. *C. elegans* research history and relevant genetics are first discussed to give context to the barriers and motivate subsequent solutions. A brief introduction to microscopy outlining two distinct approaches to observing biological specimens are then presented.

1.2 *Caenorhabditis elegans* as a Model Organism for Animal Development

In 1965, Sydney Brenner proposed the nematode *Caenorhabditis elegans* (*C. elegans*) as a model organism for development and behavior [68]. The roundworm is often studied as a model of nervous system development due its relatively simple anatomy, featuring a 302 cell nervous system in just 1000 cells in total [63, 77, 89]. Beyond simplicity, the worm has many characteristics which make it amenable to laboratory studies. Riddle writes in the seminal handbook *C. elegans II*, "This soil nematode offered great potential for genetic analysis, partly because of its rapid (3-day) life cycle, small size (1.5-mm-long adult), and ease of laboratory cultivation." The worm can be cultivated and maintained on petri dishes as they feed on *Escherichia coli* (*E. coli*) bacteria and produce hundreds of progeny per cycle [68].

Contributed research concerns biological inquiry during embryogenesis, the period of development from the one-cell stage to the embryo emerging from its eggshell. Embryogenesis occurs over the span of approximately 16 hours [77]. *C. elegans* embryogenesis has been widely studied. In particular, the complete embryonic cell lineage has been determined, i.e., all cells and their respective descendants from the one cell stage have been observed and documented [77]. The lineage is invariant worm-to-worm, that is to say that the biological codex dictating cell divisions is non-random. Moreover, the lineage was able to be derived due to cell mitosis completing approximately halfway through embryo-development [77], which coincides with the development of four bands of muscle cells. The 81 muscle cells form along the exterior of the embryo and begin contracting approximately halfway through embryogenesis [77]. The onset of muscle cells induce *twitching* in the embryo. Muscle

twitching causes sporadic bouts of movement for the remainder of development.

Rapid twitching movements complicate observation of the developing embryo. Motion blur induced by the movements causes a deterioration in image quality. Methods to automatically lineage cells [72] or perform cell tracking [17] fail when faced with abrupt transitions between images. Analysis of late-stage embryogenesis, i.e. the final hours of development prior to hatching, requires methodology for addressing and mitigating twitching. Established methods [21] concern a subset of epidermal cells know as *seam cells*.

The twenty seam cells and two associated neuroblasts form in lateral pairs along the left and right sides of the worm, resulting in eleven pairs upon hatching [77]. The Q neuroblasts appear in the final hours of development, just prior to hatching. The pairs of cells are named, anterior to posterior: *H0*, *H1*, *H2*, *V1*, *V2*, *V3*, *V4*, Q (neuroblasts), *V5*, *V6*, and *T*. Each pair's left and right cell is named accordingly; for example, *H1L* and *H1R* comprise the *H1*. The seam cells and neuroblasts together describe anatomical structure in the coiled embryo, acting as a type of "motion capture suit" outlining its body. Identification of the seam cells and Q neuroblasts reveals the embryo's *posture*. Imaging and identifying the seam cells is a challenging task, further complicated by physical and technological constraints in microscopy.

1.3 Electron & Fluorescence Microscopy

Microscopy describes the application of microscopes to view objects that would cannot be seen with the human eye. Two approaches, electron microscopy and fluorescence

microscopy, rely on similar foundational physical foundations, but have one key distinction. Electron microscopy relies on the excitation of electrons as a source of illumination, whereas fluorescence microscopy uses photon excitation as a means to illuminate the specimen. Electrons have a shorter wave-length than a photon, enabling increased spatial resolution than is possible using fluorescent methods.

Sample preparation steps for electron microscopy, including dehydration and staining, cause death of the specimen. As a result, electron microscopes are unable to produce time-lapse images of a single specimen. Embryogenesis can be observed across several embryos at different points in development, but are unable to observe one embryo fully. Electron microscopy techniques are able to achieve nanometer-scale resolution of a static sample, as opposed to fluorescence microscopy is several hundred nanometers. [1]. The connectome of *C. elegeans*, describing the map of neural connections in the brain, was completed in 1986 using electron microscopy [89]. To date, *C. elegans* is the only organism to have its connectome completed.

On the other hand, fluorophores are illuminated by light and provide molecular contrast when illuminated in fluorescence microscopy. Fluorophores emit longer wavelengths of light upon excitation via a particular wavelength, dependent on the fluorophore itself. The fluorescent microscope provides the exciting wavelength, triggering excitation of the fluorescent molecule which then emits a longer wavelength of light to be observed with the microscope detector. However, not all organisms of biological interest naturally develop fluorescent molecules. For example, the jellyfish *Aequorea victoria* contains a green fluorescent

[1]Electron microscope images of *C. elegans* captured by Nichol Thomson, one of Sydney Brenner's collaborators, inspired Brenner to pursue the worm as a model organism.

protein (GFP) which excites under a lower frequency (blue) light. Martin Chalfie earned the Nobel Prize in Chemistry by demonstrating that GFP could be expressed in *C. elegans* cells, allowing their observation in the living animal [2] [16]. The advancement further has enabled fluorescent imaging of live biological samples.

The delicate nature of fluorophores presents challenges in fluorescence microscopy. Overexcitation will cause photobleaching, an event characterized by the fluorophore becoming unresponsive and unable to fluoresce. Tempering the light dosage and imaging frequency of the exciting laser is crucial for maintaining the fluorophore's health. A tradeoff emerges; one must balance the quality of the image for the quantity of images. Images with stronger signal must be captured less frequently, whereas images of lower signal can be captured more often. Advances in fluorescence microscopy have mitigated this tradeoff. For example, light sheet fluorescence microscopy (LSFM) allows for gentle volumetric imaging by only illuminating and capturing single planar portions of the sample. The technique limits illumination to a focal plane as it is swept throughout the volumetric sample [40, 90]. Multiview lightsheet fluorescence microscopes capture orthogonal views of the sample and use image registration methods to yield a final image of higher spatial resolution than a single view can alone.

Advancements in genetic biology (Chalfie's GFP, along with subsequent fluorophores) and fluorescence microscopy (LSFM) have further enabled observation of *C. elegans* embryogenesis. Fluorescence can be selectively expressed in the seam cells and observed using light-sheet microscopy. Despite such advancements, research progress in late-stage development is

[2] Chalfie also happened to work with Brenner as a post-doctoral researcher; he used electron microscopy to study touch sensitivity in *C. elegans* mutants.

stymied by embryonic twitching. Mathematical and computational contributions presented address seam cell identification in an effort to promote relevant biological discovery. Although methods were motivated by seam cell identification, contributed methods are applicable to other combinatorial optimization problems and object tracking tasks.

1.4 Motivation

Presented research contributes to a lineage of biological research centered about *C. elegans*. The organism has led to numerous discoveries concerning development, particularly neurodevelopment. Many advancements were birthed with the collaboration of microscopists and geneticists. As noted above, it was Thomson's electron micrographs which first inspired Brenner's pursuit of of *C. elegans*. Biology has come into a renaissance under a new collaboration with mathematicians and computer scientists, enhancing the field of computational biology. The partnership has expanded the imagination of biologists as their imaginations can now wander further; experiments can address more factors and the data can be both analyzed in new ways and more efficiently parsed. The outcome is not only more, but also higher quality scientific research being pursued.

Biological research goals addressed are at the precipice of *C. elegans* embryo-development. Two major questions are pursued, both pertaining to late-stage development. The first goal concerns understanding how cells form tissues, organs, and organ structures as the embryo develops. The project necessitates the cataloguing of cells across time in the embryo, i.e. building a 4D (3D + time) atlas of tissue development; muscular twitching has complicated relevant analyses as cells cannot feasibly be imaged and tracked during twitching bouts.

The seam cells are used as fiducial markers to mitigate embryo displacement. Identifying the seam cells with no temporal context emerges as a challenging modeling task. The second research project focuses directly on nervous system development, particularly on embryonic "behavior." How and when the nervous system forms and gains control of motor function is not currently understood. The final hours of development contain the moments in which a bumbling embryo matriculates into a neuronally controlled organism. Observing behavioral development requires direct imaging and tracking of seam cells despite twitching bouts. The seam cells together approximate the coiled embryo's *posture*, analysis of time-lapse posture transformations serves as a test-bed for understanding behavior.

Both of these research questions push the boundary of fluorescence microscopy (Chapter 1.3); limitations in optical physics and its current application prohibit simple solutions to each research project. The aforementioned tradeoff between image quality and frequency of imaging leaves solutions to both projects requiring expert intervention; existing computational methods achieve poor quality in each environment. Seam cell identification is a key step of discovering new biological phenomena. However, the imaging paradigm of each project is at one end of the quality-quantity tradeoff. Imaging approaches for the tissue development project prioritize quality, as many cell nuclei within the embryo must be distinguishable throughout late-stage development. As a result, images are captured minutes apart to ensure health of the embryo. On the other hand, studying behavior requires a high capture rate to observe subtle patterns of movement. Images are captured at the maximum capture rate of the volumetric microscope, but at the expense of image quality.

Seam cell nuclei were imaged and used as an intermediate step towards the biological discovery in each research project. More specifically, fluorescent promoters were genetically

introduced into the embryos such that when seam cell nuclei expressed green light when exposed to blue light. The bright spheroids appear in an otherwise dark image volume, akin to stars in the night sky. Furthering the analogy, stars observed together in imagined patterns form constellations. The posture is in this sense its own constellation inside the eggshell, but the embryo's movements cause the constellation to change shape. Posture identification is akin to finding the same constellation (posture) as it changes shape throughout late-stage development due to contortions and elongation. Each seam cell plays a *role* in comprising the posture. Experts know a seam cell's identity by contextualizing positions of other visible seam cells. The posture is an abstraction of cells together forming a shape just as a constellation is an abstraction of its comprising stars. The task in both cases is to output a set of pairings between observed bright spheroids and seam cell identities in a given image volume. Each seam cell is found only once per image and each imaged nucleus belongs to a seam cell. These one-to-one constraints are common in many modeling tasks, so much so that the field of *assignment problems* comprises optimization problems and algorithms centered around solving such problems.

Assignment problems describe a type of discrete optimization problems. In particular, they are binary integer programs with the aforementioned one-to-one constraints. They describe the optimization problem concerning the matching of objects between two disjoint sets. Posture identification was cast as assignment problem in which seam cell nuclei, an observed set, were matched to seam cell identities, an abstract set. The observed nuclei themselves can be represented as points in 3D space. Assignment problem algorithms can be applied to pair points to seam cell identities. However, seam cell identification is markedly different from assignment problems encountered in literature for myriad reasons

to be described in later chapters. Assignment problems themselves can be split into two broad categories. The key juxtaposition concerns the existence of an underlying *correct* matching.

The first class of assignment problems is truer to the origins in the field; they are complex tasks in which even an expert cannot reasonably deduce an optimal answer. Logistics, resource allocation, vehicle routing, and economic planning problems all need advanced algorithms to identify optimal decisions. For example, the travelling salesman problem describes a canonical discrete optimization in this subfield. The problem concerns finding an optimal (often minimum distance) tour through a set of locations. As an example, delivery drivers start at a warehouse and have a list of locations in which to drop off goods. The driver wants to end at the warehouse and stop at each location only once in the fastest way possible. Assignment problem methodology is applied ubiquitously to logistics problems to maintain efficient global supply chains. The key characteristic of this class of problems is that the optimal answer is not known; an algorithm needs to find the *optimal* answer which is then taken as the *correct* answer.

On the other hand, there exist problems in which a *correct* answer exists and the goal is to model the task such that the optimal answer is the correct answer. If possible, the end result is an autonomous system which can yield accurate predictions and improve efficiency in data analysis pipelines. Point-set matching problems often fall into this category. A human can observe a transformation among points, such as an image rotating, and match fiducial markers between images. Tasks such as object tracking or registration have this element of *correctness* to them. Posture identification fits into this category; each posture can be confirmed by experts; the goal is to build a method which can automatically identify

the embryo's posture (identify all seam cells) in a given image volume.

Assignment problems of the second category can be viewed as a supervised learning task. Point-set matching problems operate using an underlying *model* of correspondence. For example, a model based upon frame-to-frame displacement is used in some multiple object tracking (MOT) methods. This is to say that objects which are close between frames are likely the same object; the optimum is the association which minimizes total displacement. Models *predict* associations which can then be compared to expert annotations for evaluation. Effective models then must be powerful enough to use underlying patterns, but flexible enough to generalize to new instances. The bias-variance tradeoff comes to mind. An appropriate model will have the necessary capacity (minimum bias) to correctly associate points between sets while remaining as simple as possible to minimize prediction variance. Weak models will fail to capture relevant patterns while overly complex models will not generalize to unseen instances of the task at hand. Point-set matching models are comprised of features which are measurements of the objects themselves. While many features use the coordinates of the points, other features may be based on some facet of the object's appearance, such as shape, color, and size.

The comparison between point-set matching tasks and supervised learning is imperfect. Assignment problem constraints are unable to be enforced in current supervised learning optimization processes. While structured output methods exist, the uniqueness constraint is particularly difficult to maintain [80]. Supervised learning has nonetheless been attempted in point-set matching problems [14]. Model prediction in a typical supervised learning process does not require further optimization. However, prediction in an assignment problem requires *solving* the assignment problem. Searching the permutation space (there

are $n!$ permutations of a set of size n) can be computationally costly depending on the model and approach. Seminal research unites the two fields, demonstrating a framework for gradually updating a graphical model's parameters and showed parametric models are more accurate than parameter-free models [14].

Types of features used in an assignment model directly relate to the computational burden of finding a solution. Increasingly intricate models require more computation to solve; the steep increase in computation for complex models prohibits application to any exceedingly challenging point-set matching task. Recent point-set matching methods in the literature are applied on benchmark datasets which feature relatively simple transformations [27, 96, 97]. Posture identification is notably more challenging as the embryo may entirely reposition between images. Sudden twitching movements in the flexible embryo present challenge even in the high capture-rate imaging paradigm. Predicting embryonic posture is not comparable to matching points between images of a house during a slight rotation.

Posture identification as a barrier to cutting-edge biological research motivates this writing. The problem lies at the intersection of statistics, discrete optimization, and developmental biology motivated my contributions to the fields of combinatorial optimization and multiple object tracking. The exploration of research questions arising in both paradigms yielded new methods and mathematical methods while assisting in discovering new biological phenomena.

1.5 Research Questions

Research questions explored in the subsequent chapters are driven by the pursuit of identifying posture in each of the two imaging approaches. Accurate posture identification was the foremost goal in both explorations. Differences in imaging conditions led to the adaptation of established methods in assignment problem literature to solve posture identification in each scenario.

Posture identification for the tissue development project was investigated as a supervised learning oriented assignment problem with intent to create a framework for other challenging object identification tasks. Established methods for point-set matching were unable to be generalized to highly interdependent features nor were they able to guarantee exact solutions [27, 96]. The pursuit of automatic seam cell identification would require stronger models capable of describing relationships between cells. Semi-automatic posture identification stood as a secondary goal if the task could not be reliably performed automatically.

The higher capture rate data yielded a different set of concerns. Posture tracking was able to use locations at the preceding frame to better guide identification. Similarities between problems encouraged design of an interdependent tracking paradigm. Here, we investigated the efficacy of correlated object movement into multiple object tracking methods. Established methods were more easily applicable to behavioral analysis data, but two notable facets complicated posture tracking. First, the rapid image capture rate is offset by a low laser dose at each image, yielding images in which seam cell nuclei were not easily detectable. Additionally, aforementioned muscular twitching causes random bouts of motion. The adversarial tracking landscape was balanced by the correlated seam cell

motion.

1.6 The n-adic assignment problem

Assignment problems were integral to explored research questions and contributions. The combinatorial optimization problems are defined by their carefully constructed constraints. They are formulated as binary integer programs with one-to-one constraints to ensure solutions represent a valid "matching" of objects between sets. The objective formulation directly describes relationships between objects. For example, the linear assignment problem encodes independent object-to-object relationships while being solvable in worst-case polynomial time [35, 39]. Higher order assignment problems such as the quadratic, cubic, or quartic assignment problems are $\mathcal{NP}$-complete [70]. As a result, published methods mostly concern heuristic methods applied to lower order forms. The highest degree assignment problem, the n-adic assignment problem, had not yet been solved. The more general n_1 to n_2 ($n_1 \leq n_2$ rectangular assignment problem will be explored. The permutation matrix space Π is expanded to the assignment matrix space $\mathcal{X}$:

$$\mathcal{X} = \{X \in \{0,1\}^{n_1 \times n_2} : \forall j, \sum_{i=1}^{n_1} x_{ij} \leq 1, \forall i \sum_{j=1}^{n_2} x_{ij} = 1\}$$

While the linear assignment problem uses a cost matrix, and the quadratic assignment problem a 4D cost tensor, the n_1-adic assignment problem requires a degree $2n_1$ tensor to store dissimilarities arising from a full n_1 to n_1 assignment: $(l_1, l_2, \ldots, l_{n_1}) \mapsto (l'_1, l'_2, l'_3, \ldots, l'_{n_1})$. The n_1-adic assignment problem is cast as *hypergraph matching*. The number of vertices aligned by the most comprehensive hyperedge defines the degree of a hypergraph. Maximum

degree hypergraphs with hyperedges composed of all n_1 vertices yield the most comprehensive point-set matching function possible. Then, for a given assignment matrix $X \in \mathcal{X}$, the hypergraph matching objective can be expressed using n_1 dissimilarity tensors of dimension $2, 4, \ldots, 2d, \ldots, 2n_1$, each measuring dissimilarity between degree d hyperedges, respectively. Define $\mathbf{Z}^{(d)}$ as the tensor mapping the dissimilarity for the degree d hyperedges. The hypergraph matching objective is expressed in Eq 3.2.

$$
\begin{aligned}
f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(n_1)}) = & \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \mathbf{Z}^{(1)}_{l_1 l'_1} x_{l_1 l'_1} + \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \sum_{l_2=l_1+1}^{n_1} \sum_{l'_2=1}^{n_2} \mathbf{Z}^{(2)}_{l_1 l'_1 l_2 l'_2} x_{l_1 l'_1} x_{l_2 l'_2} \\
& + \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \sum_{l_2=l_1+1}^{n_1} \sum_{l'_2=1}^{n_2} \sum_{l_3=l_2+1}^{n_1} \sum_{l'_3=1}^{n_2} \mathbf{Z}^{(3)}_{l_1 l'_1 l_2 l'_2 l_3 l'_3} x_{l_1 l'_1} x_{l_2 l'_2} x_{l_3 l'_3} + \ldots \\
& + \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \ldots \sum_{l_{n_1}=l_{n_1-1}+1}^{n_1} \sum_{l'_{n_1}=1}^{n_2} \mathbf{Z}^{(n_1)}_{l_1 l'_1 \ldots l_{n_1} l'_{n_1}} x_{l_1 l'_1} \ldots x_{l_{n_1} l'_{n_1}}
\end{aligned}
$$

Hypergraph matching allows for the modeling of intricate point-set matching problems through high multiplicity assignment objective function formulations. The $\mathbf{Z}^{(d)}$ dissimilarity terms measure degree d hyperedge dissimilarity comprising d simultaneous vertex assignments. The range in assignment problem objective complexity from d=1 to d=n_1 trades off model capacity for increased computation. The traditional linear assignment problem (d=1) is solvable in polynomial time [39], but treats points between sets independently. Existing graphical methods (d=2) and hypergraphical methods (d>2) rely on approximate searches and do not generalize to high degree formulations of Eq 3.2. We explore a method to find globally optimal solutions to hypergraph matching problems of arbitrary degree, allowing for the modeling of intricate point-set matching tasks.

1.7 Contributions

Contributions encompass both specialized tools for posture identification and posture tracking as well as general methodology for underlying optimization problems. Effective applied mathematics research lies somewhere between each endpoint; new methods ought to solve the inspiring problem but also be flexible enough to be applied elsewhere (a macro-level bias-variance tradeoff). Mathematical methodology built with a rigorous foundation is malleable and can be adjusted for applications other then how it was originally intended.

Chapter 3 explores the modeling and solving of the n_1-adic assignment problem, i.e. solving an assignment problem in which all objects between sets interact. The task was phrased as hypergraph matching where the hypergraph features a degree n_1 hyperedge. Existing branch-and-bound techniques were adapted to exactly solve the resulting assignment problem. The contributed branching algorithm, Exact Hypergraph Matching (EHGM), exactly solves the n_1-adic assignment problem. The algorithm is the first in the field to do so. EHGM was used in conjunction with biologically inspired hypergraphical models to identify posture in low temporal resolution imaging. Presented methods improved posture identification accuracy over existing graphical methods while also acting as a frame/.work for approaching similarly challenging point-set matching tasks.

High spatial resolution is necessary for tracking nuclei of other cells in late-stage development. Posture identification enables a change of basis from the coiled ("twisted") to the straightened ("untwisted") space. The posture-based remapping removes variation in cell positioning attributable to the embryo moving, leaving only the movement *within* the embryo. Chapter 2 describes the untwisting process, and proposes a semi-automatic cell

tracking interface. Tracking inter-embryo cell movement was modeled as a multiple object tracking task. Results show traditional methods accurately tracked cells when detections were accurate; however, densely packed nuclei were difficult to distinguish reliably. An iterative tracking paradigm was built into a graphical user interface (GUI) for users to semi-automatically track cell nuclei.

Chapter 2 explores the posture tracking problem. The rapid image capture rate requires a low dose of light when capturing each image to avoid photobleaching. The low spatial resolution, high temporal resolution volumetric images of seam cell nuclei served as input to track posture frame-to-frame over the last five hours of embryo-development. The task was modeled as a multiple object tracking (MOT) problem. Multiple hypothesis tracking (MHT), a leading method in the field, was augmented via hypergraphical modeling to more accurately track posture despite sporadic bouts of twitching and imperfect detections. Multiple hypothesis hypergraph tracking (MHHT) was inspired by posture tracking, but is applicable to other MOT problems featuring interdependence between objects.

A subsequent analysis of posture transformations revealed patterns of emergent *behavior* in the embryo. Mutants lacking a certain gene failed to express mature behavior. The posture libraries were posited as a corpus for exploration of behavior while contributed methods enable the curation of further behavioral data.

Contributions across both posture identification and posture tracking can best be summarized as the development and application of interdependent assignment problem models and accompanying algorithms. The mathematical rigor of the methods allows further tuning, adaptation, and application to not only other tracking problems but other combinatorial optimization problems.

Chapter 2: Mathematical Preliminaries

Presented contributions can be summarized as methods to perform object identification, specifically cell identification in time-lapse fluorescence microscopy images. Assignment problems comprise the mathematical foundation of object identification. Then, multiple object tracking (MOT) methods use these optimization frameworks to solve the real-world scenario of maintaining information about a unique set of objects in a discrete-time sequence of images. However, the cell nuclei themselves must first be detected in such image sets. The aforementioned tradeoffs in fluorescence microscopy (Chapter 1.3) can make the problem notably challenging, presenting a powerful initial barrier to performing MOT in fluorescence microscopy images. A review of methods to perform object detection as well as an overview of MOT will follow a lengthy exposition of assignment problems.

2.1 Assignment Problems

Assignment problems describe how to align objects within one set to objects of another. This scenario is ubiquitous in day-to-day life. The *marriage problem* is a canonical example; young men and young women are to be matched in a way which yields the most marriages. Another application concerns the matching of employees to roles within a company. Generally, let sets U and V each be of size n. An assignment defines a

bijective mapping of between sets; assignments are often expressed as permutations φ of the first n natural numbers: $1, 2, \ldots, n$, where object index i of U is mapped to object index $j = \varphi((i)$ of V. The permutation $\varphi = (\varphi(1), \varphi(2), \ldots, \varphi(n))$ defines one of the $n!$ possible permutations.

Each assignment can be expressed via a permutation matrix. First define the binary variable x_{ij}:

$$x_{ij} = \begin{cases} 1 & \text{if } j = \varphi(i) \\ 0 & \text{otherwise} \end{cases}$$

The matrix $\mathbf{X}_\varphi = [|x_{ij}|]$ corresponds to this unique permutation of the n rows of the size n identity matrix, $\mathbf{I}_n$. The permutation matrix space Π comprises the set of all permutation matrices of size n. Assignment problems operate over the domain Π:

$$\Pi = \{\mathbf{X} | \mathbf{X} \in \{0,1\}^{n \times n}, \sum_{i=1}^{n} x_{ij} = 1, \sum_{j=1}^{n} x_{ij} = 1\} \tag{2.1}$$

Assignment problems can also be expressed using bipartite graphs. A bipartite graph $G = (U, V, E)$ combines the two disjoint object sets U and V with an edge set E. Edges $e \in E$ must exclusively connect objects between sets; that is, any edge cannot contain both endpoints in the same set. Matching describes the task of finding a sparse subset $M \subseteq E$ such that each vertex coincides with one edge [13]. The task of finding an optimal assignment can then be phrased as identifying an optimal match between nodes of a bipartite graph.

Assignment problems describe a mathematical framework for organizing and expressing

matchings. Two questions naturally arise: *How do we define an optimal assignment* and *How do we find an optimal assignment?* As in any subfield of optimization, the answer to the latter depends on the former.

2.1.1 The Linear Assignment Problem

The linear assignment problem (LAP) presents the simplest model of an assignment problem. A cost matrix $\mathbf{C}$ specifies the cost of assigning object i of U to object j of V via the entry c_{ij}. The total cost of a permutation φ, given $\mathbf{C}$, can be expressed $\sum_{i=1}^{n} \mathbf{C}_{i\varphi i}$. The optimization problem can then be expressed:

$$\min_{\varphi \in S_n} \quad \sum_{i=1}^{n} c_{i\varphi i} \tag{2.2}$$

The permutation matrix formulation of the LAP uses binary optimization variables x_{ij}. The constraints ensure each $n \times n$ matrix is an assignment matrix.

$$\begin{aligned} \min \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_{ij} = 1 \qquad i = 1, 2, \ldots n \\ & \sum_{j=1}^{n} x_{ij} = 1 \qquad j = 1, 2, \ldots n \\ & x_{ij} \in \{0, 1\} \end{aligned} \tag{2.3}$$

The LAP follows the form of a linear program. As such, traditional methods for solving linear programs can be applied to the LAP. However, specific algorithms were developed just for the LAP as well.

2.1.1.1 Solving the LAP

Algorithms for solving the LAP rely on a key observation regarding the domain Π. The continuous relaxation of Π yields the *doubly stochastic* matrix space, $\mathcal{D}$ (Eqn 2.4. Here, the rows and columns of matrices $X \in \mathcal{D}$ still sum to one, but entries are relaxed to the closed interval: $x_{ij} \in [0, 1]$. It is evident that every permutation matrix is also a doubly stochastic matrix, i.e., $\Pi \subset \mathcal{D}$. The set $\mathcal{D}$ defines the assignment polytope P_A. Birkhoff showed that each vertex of P_A corresponds to a unique permutation matrix $X \in \Pi$ [9]. As a result, every doubly stochastic matrix can be written as a convex combination of permutation matrices [13].

$$\mathcal{D} = \{\mathbf{X} | x_{ij} \geq 0, \sum_{i=1}^{n} x_{ij} = 1, \sum_{j=1}^{n} x_{ij} = 1\} \tag{2.4}$$

Birkhoff's result is a step towards relating LAP, a combinatorial optimization problem, with a traditional linear program. The fundamental theorem of linear programming states that optimal solutions to a linear objective lie on the vertices of the convex polygonal region formed by the linear constraints. The two results together pave a strategy for solving the LAP. Optimization strategies in the doubly stochastic space will converge to a vertex of the assignment polytope, which Birkhoff showed to be a permutation matrix [13].

Harold Kuhn's Hungarian algorithm provided the first polynomial time solution to the LAP [13, 39]. Kuhn's algorithm uses both the primal (Eqn 2.3 and dual (Eqn 2.5) to identify a global minimum; the method is $\mathcal{O}(n^4)$. Two sets of dual variables v_j and u_i, $i, j = 1, 2, \ldots, n$ are used to find a solution which achieves complementary slackness [39].

The Jonker-Volgenant method improves the process to $\mathcal{O}(n^3)$ [35].

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} u_i + \sum_{j=1}^{n} v_j \\ \text{s.t.} \quad & u_i + v_j \leq c_{ij} \qquad i, j = 1, 2, \ldots n \end{aligned} \tag{2.5}$$

Auction algorithms frame the bipartite graph matching task through the lens of an auction. The paradigm iteratively commits to matchings until a global optimum is achieved. The process is noted to be more efficient when the nature of the matching process is able to be split into disjoint regions, such as in multiple object tracking applications [24]. With auction algorithms, a slackness parameter ϵ is used to decrease computation at risk of converging upon a suboptimal solution [7]. The Jonker-Volgenant-Castañon (JVC) algorithm modifies the Jonker-Volgenant algorithm to include an auction step at initialization [26]. The JVC algorithm is currently used as the most efficient globally optimal algorithm for the LAP [1] [24].

The LAP stands as one of the most prominent and versatile linear programs. The model's simplicity allows for a breadth of applications, while the existence of polynomial time algorithms ensures scalability and real-time use. However, a linear objective is too simplistic for certain applications. Indeed, all models aim to simplify or approximate an entangled real-world scenario, but the LAP is unable to account for any level of dependence in assignments. Linearity of the optimization variables implies the model's cost of assigning objects between sets is independent of other simultaneous assignments.

[1] The JVC algorithm also allows for rectangular cost matrices.

2.1.2 Quadratic Assignment Problem

The quadratic assignment problem (QAP) was introduced by Koopmans and Beckmann [38] in context of locating "indivisible economic activities." The model was inspired by the problem of placing facilities in a manner which maximizes profit. The optimal placement of facilities depends on relationships between local pairs of facilities in how one facility uses what the other produces. The problem is *quadratic* in nature due to this pairwise dependence; a product of the binary optimization variables appears in the objective function to quantify the relationships. The Koopmans and Beckmann formulation uses two matrices, distance between facilities A and flow between facilities B in their inspiring example, to characterize the "indivisibility" problem. The travelling-salesman problem can be expressed as a Koopmans-Beckmann form QAP, evidence of the applicability of the QAP to combinatorial optimization tasks [43].

Lawler proposed a more general expression of the QAP, in which the Koopmans-Beckmann form is a special case [43]. The Lawler formulation extends the LAP objective with a degree four tensor $d_{ijkl} = a_{ik}b_{jl}$ to represent "the cost of transportation from plant i at location k to plant j at location l."

$$\begin{aligned} \min \quad & \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} d_{ijkl}x_{ik}x_{jl} + \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ik}x_{ik} \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_{ik} = 1 \qquad k = 1, 2, \ldots n \\ & \sum_{k=1}^{n} x_{ik} = 1 \qquad i = 1, 2, \ldots n \\ & x_{ik} \in \{0, 1\} \end{aligned} \tag{2.6}$$

Lawler's QAP formulation can also be viewed as an extension of the LAP (Eq 2.3). The quadratic term d_{ijkl} can be used to model the cost arising from the joint assignment of i to k *and* j to l, where i and k are in the first set and j and l the second set. The formulation can be naturally applied to nodes and edges in graphs. Graph matching, or more formally the graph isomorphism problem, describes the task of finding an optimal alignment between vertices of a graph subject to vertex-to-vertex and edge-to-edge costs. The linear portion of the Eq 2.6 objective measures the dissimilarity of vertices i and j, while the quadratic portion measures the dissimilarity of the edge formed by nodes i and k and the edge formed by nodes j and l.

2.1.2.1 Solving the QAP

Finding an exact solution to the QAP is an $\mathcal{NP}$-hard problem. That is, unless P=NP, there does not exist a polynomial time solution to exactly solve the QAP. Additionally, the problem is *strongly NP*-hard; there does not exist a polynomial time approximation algorithm within a constant factor of the optimal solution [70]. As a result, methods for solving the QAP can be stratified into two categories: heuristic methods and exact methods. Heuristic methods approximate a solution with no bound on optimality due to the strong $\mathcal{NP}$-hardness. Exact methods, on the other hand, guarantee a globally optimal solution, but have worst-case exponential computational complexity. A review will highlight leading heuristic methods as well as branch-and-bound, a seminal paradigm in combinatorial optimization.

Heuristic methods for solving the QAP rely on the continuous relaxation of Π to

the doubly stochastic space $\mathcal{D}$. However, the objective is quadratic and non-convex. Approximate algorithms may converge to suboptimal solutions.

Early approaches approximating the objective with a linear term and gradient based methods are used to converge to a solution [32]. Spectral methods aim to approximate the adjacency matrix by calculating the leading eigenvector of the affinity (or dissimilarity) matrix storing edge-to-edge and node-to-node matching weights [45]. Path-following strategies using a weighted combination of convex and concave relaxations have emerged at the forefront of heuristic methodology [48, 84, 93]. A final projection step via an LAP yields a permutation matrix solution in all approaches.

Exact methods rely on implicit enumeration of the search space S_n to find an optimal assignment. Branch-and-bound is a paradigm originally developed to solve the travelling salesman problem, a type of quadratic assignment problem [41, 47]. The methods are among the most efficient, although there does not appear to be consensus on a generally best paradigm for exactly solving quadratic assignment problems. Branch-and-bound methods recursively commit partial assignments and solve successive subproblems within the search space, S_n. The algorithm iteratively partitions the search space while bounding the optimum at each branch. At each step the method prunes branches which cannot contain the optimizer. Convergence occurs when only feasible assignments achieving a global optimum remain. The $\mathcal{NP}-$hardness of the QAP implies convergence occurs only after implicit enumeration of S_n.

2.1.3 N-adic Assignment Problems

Lawler noted the extension to N-adic assignment problems in his formulation of the QAP [43]. Cubic and quartic (biquadratic) assignment problems extend Lawler's objective to include degree three and four products of the assignment variables, respectively. Such assignment problems and higher order objective formulations are also $\mathcal{NP}$-hard, as they are lower bounded in complexity by the QAP [70].

2.1.3.1 Hypergraph Matching

Higher order assignment problems can be applied to model *hypergraph matching* just as the QAP is applied to model graph matching. Hypergraphs extend the definition of a graph to include hyperedges which can specify relationships among an arbitrary number of vertices. Hypergraph matching then concerns finding an optimal vertex correspondence between pairs of attributed hypergraphs. The number of vertices aligned by the most comprehensive hyperedge defines the degree of a hypergraph. For example, the cubic assignment problem can be written as hypergraph matching with degree three hyperedges which each specify a relationship between vertices a, b, c in the first hypergraph with vertices d, e, f in the second. Hyperedges are able to express higher degree joint relationships while edges can only express a bivariate relationship.

2.1.3.2 Solving Hypergraph Matching

Heuristic hypergraph matching methods follow from existing graph matching algorithms. In particular, spectral methods for solving Lawler's QAP (Eq 2.6) have been extended to

solve hypergraph matching. Duchenne et al. adapt Leordeanu's work to obtain a rank-1 approximation of the affinity tensor via higher order power iteration [27, 45]. However, calculating the affinity tensor is computationally prohibitive, especially for higher degree hypergraphs. Simplifying assumptions such as super-symmetry and sparseness are used with sampling methods to build large affinity tensors [27, 94]. Chertok and Keller propose similar methodology, but instead unfold the affinity tensor and use the leading left singular vector to approximate the adjacency matrix [18]. All such methods operate outside the permutation matrix space. The Hungarian algorithm or similar binarization step is used to yield a valid assignment, e.g. as in [45].

Heuristic methods rely on computing an affinity tensor prior to each proposed method in [18, 27, 57, 91, 95]. This step serves as a computational barrier to higher degree hypergraph matching as calculating the affinity tensor scales exponentially with the size of the largest hyperedge [27]. Methodologies in previous work justify their strategies as the hypergraphs are limited to degree three hypergraphs allowing the description of angles between triplets of points [18, 19, 27, 57, 95]. Such methods are unable to generalize to high degree hypergraph matching tasks while also providing no certainty as to the optimality of the returned correspondence.

2.1.4 Multidimensional Assignment Problems

Assignment problems discussed thus far have described identifying an optimal correspondence of points between two disjoint sets. The optimization objectives allow for relationships between: individual points, pairs of points, and so on to relationships between all points

within a set to all points within the other. However, the relationships are still measured between just two point-sets.

Another type of assignment problem allows flexibility in the number of sets in which points are matched. The multidimensional assignment problem (MAP) extends the traditional assignment problem formulation to match points between $L > 2$ sets [60]. Assume points are to be matched between L=3 sets with $p \leq q \leq r$. The MAP with linear cost tensor $\mathbf{C}$ is expressed via Eq 2.7.

$$\begin{aligned}
\min \quad & \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k=1}^{r} c_{ijk}x_{ijk} \\
\text{s.t.} \quad & \sum_{i=1}^{p}\sum_{j=1}^{q} x_{ijk} = 1 \qquad k = 1, 2, \dots r \\
& \sum_{i=1}^{p}\sum_{k=1}^{r} x_{ijk} = 1 \qquad j = 1, 2, \dots q \\
& \sum_{j=1}^{q}\sum_{k=1}^{r} x_{ijk} = 1 \qquad i = 1, 2, \dots r \\
& x_{ijk} \in \{0, 1\} \qquad \forall\, i, j, k
\end{aligned} \tag{2.7}$$

The generalization to three sets can be intuitively understood through the lens of a standard LAP (Eq 2.3). Assume p workers are to be assigned to q jobs; this problem can be solved with the LAP. However, if we were to assume that employees change jobs at r different points in time, then the MAP with L=3 can model the task appropriately. The MAP with $L > 2$ is $\mathcal{NP}$-hard [36]. Pierskalla's solution adapts branch-and-bound methodology to solve LAPs within the solution space [60].

2.2 Cell Detection & Tracking in Fluorescence Microscopy

2.2.1 Nucleus Detection

Nuclei of certain cells can be accurately targeted to emit light during imaging via fluorescent proteins. The practice enables the observation of subgroups of nuclei pertaining to organ systems and cellular processes *en vivo*. The resulting fluorescent images can be interpreted as *signals* in which the desired objects emit a Gaussian-like (also referred to as *blob*-like) intensity pattern about the center of each object. Methods to detect nuclei are a subset of image segmentation techniques which are built using the assumed Gaussian profile of objects.

Image segmentation aims to partition an image into disjoint regions formed by sets of connected pixels. Image segmentation methods can be split into two groups: instance segmentation and semantic segmentation. Semantic segmentation assigns a class label to each pixel in an image (or voxel in an image volume), while instance segmentation identifies clusters of pixels defining an instance of an object within an image. Semantic segmentation methods serve as an initial step towards identifying instances of objects of interest. Postprocessing routines such as a connected components analysis will cluster pixels together to yield disjoint objects.

Traditional methods in segmentation for fluorescence microscopy leverage the Gaussian-like profile of imaged objects. A blob describes a cluster of pixels belonging to one object. A Gaussian intensity profile approximates each fluorescent blob-like object; roughly spherical or ellipsoidal objects exhibit peak intensity near the center and lower intensity towards

the edges. Locating homogeneous objects such as cells, cell nuclei, or particles is then equivalent to detecting blobs in an image. Both traditional and modern strategies are evaluated to perform seam cell nuclei detection. Traditional methods rely on homogeneous blob-like structures while modern deep learning based methods use a corpus of training data with a highly parameterized (millions of parameters) graphical model. Table 2.1 lists each method and its implementation for seam cell nuclei detection.

Method	Citation	Implementation
IFT-Watershed	[29]	[49]
LoG-GSF	[46, 51]	[79]
Wavelet	[58]	[25]
Mask-RCNN	[33]	[82]
3D U-Net	[98]	[20]
Stardist 3D	[86]	[85]

Table 2.1: Multiple strategies are evaluated to perform seam cell nuclei detection. Both traditional methods (IFT-Watershed, LoG-GSF, Wavelet) and modern deep learning based methods (Mask-RCNN, 3D U-Net, Stardist 3D) are listed with implementation citations.

Three traditional segmentation approaches were applied: Image Foresting Transform Watershed (IFT-Watershed) [29], Laplacian of Gaussian with Gaussian shape fitting (LoG-GSF) [46, 51], and a Wavelet based method [58]. The IFT-Watershed algorithm uses a bottom-up approach to merge maxima from a Euclidean distance transform into disjoint regions [29]. LoG-GSF first applies the Laplacian of Gaussian, a staple method for blob detection [46]. Resulting spots are filtered according to a quadratic fitting scheme [51]. The Wavelet method uses a wavelet transform decomposition across multiple scales to identify bright spots [58]. These methods use few parameters and are able to capture homogeneous bright spots effectively.

Neural networks are parametric graphical models that leverage large amounts of

annotated data to *learn* a compositional functional relationship between inputs and outputs. Convolutional neural networks (CNNs) are a specific type of neural network better able to process both planar and volumetric image data; these models yield the best results in biological image segmentation [69, 86, 98]. Fully convolutional networks (FCNs) are a subclass of CNNs that output an image or volumetric image the same shape as the input, known as the encoder-decoder network architecture [50]. Image features are extracted from the image, and iteratively downsampled and processed throughout the network to expand field of view while learning more abstract representations. The U-Net demonstrated how FCNs could revolutionize semantic segmentation in microscopy [69]. The RCNN extended the FCN to perform instance segmentation with the region proposal network of [33, 82]. However, the Mask-RCNN is currently only able to process images, not volumes. Semantic segmentation is possible on image volumes via volumetric convolutions. The 3D U-Net demonstrated the effectiveness of stacked images as context for segmenting microscopy images [98]. More recently, *Stardist 3D* combines elements of a volumetric FCN, such as the 3D U-Net, but with a focus on identifying disjoint objects in fluorescence microscopy [85, 86]. The key contribution of *Stardist 3D* is a processing algorithm which inscribes convex polyhedra into detected blobs with the goal of separating close or touching objects.

2.2.2 Multiple Object Tracking

Multiple object tracking (MOT) is a fundamental task in computer vision. MOT is defined as maintaining information about a dynamic set of objects throughout a sequence of discretely sampled images. MOT is an intermediate problem that is often a first step

to gather contextual information for more complex types of analysis or decision making, particularly in biomedical imaging. Contributions summarized in Chapter 1.7 can each be seen as MOT methods applied to track certain sets of cell nuclei in embryonic *C. elegans*.

MOT is often expressed as two sequential steps: detection and association. Detection describes the process of obtaining measurements from images, while association refers to matching aforementioned measurements to unique object tracks. Accurate detections are crucial for an effective association step; however, moving objects in close proximity are difficult to track even with perfect detections. Frame-to-frame methods, such as the Global Nearest Neighbor (GNN) filter, are prone to failure when faced with erroneous measurements and moving objects in close proximity. Multiple hypothesis tracking (MHT) is a leading method for association, particularly for applications in fluorescence microscopy in which homogeneous object paths may intersect [17]. MHT uses detections from future frames to disambiguate challenging association decisions [10, 64].

The GNN filter serves as the canonical tool for data association in MOT. The method describes a linear program (LP) in which measurements are uniquely associated to tracks. Distances between tracks and measurements are used to find a globally optimal pairing between points of the two sets. The GNN is expressed as the linear assignment problem (LAP) (Chapter 2.1.1). The assignment constraints enforce each track $i = 1, 2, \ldots, n$ being uniquely matched to one detection $j = 1, 2, \ldots, n$. The data association problem leverages the LAP to perform MOT.

Define $\mathbf{Z}^{(t)} = [\mathbf{z}_1^{(t)}, \mathbf{z}_2^{(t)}, \ldots, \mathbf{z}_n^{(t)}]$ as the states of each object $i = 1, 2, \ldots, n$ at time $t = 1, 2, \ldots, T$. State $\mathbf{z}_i^{(t)}$ describes the center position of object i at time t. Similarly, define $\mathbf{O}^{(t)} = [\mathbf{o}_1^{(t)}, \mathbf{o}_2^{(t)}, \ldots, \mathbf{o}_{m^{(t)}}^{(t)}]$ as the set of measurements at frame t, indexed $j = 1, 2, \ldots, m^{(t)}$,

$t = 1, 2, \ldots, T$. The detection step generates sets $\mathbf{O}^{(t)}, t = 1, 2, \ldots, T$, while the association step concerns using the detections to update tracks $\mathbf{Z}^{(t)}, t = 1, 2, \ldots, T$.

The GNN cost matrix $\mathbf{C} \in R^{n \times (m(t)+n)}$ (Eq 2.8) specifies costs for associating measurements $j = 1, 2, \ldots, m^{(t)}$ to states $i = 1, 2, \ldots, n$. The matrix comprises two blocks of sizes $n \times m(t)$ and $n \times n$. The first block measures the euclidean distance between track $\mathbf{z}_i^{(t-1)}$ and detection $\mathbf{o}_j^{(t)}$ while entries in the second block consists of costs of non association known as *gates*. Each gate $d_i^{(t)}$ allows for track i to receive no measurements at time t. In the context of the GNN filter, the gate specifies a distance radius about each track $\mathbf{z}_i^{(t-1)}$ in which measurements $\mathbf{o}_j^{(t)}$ must reside in order to associate to track i.

$$\mathbf{C}^{(t)} = \begin{bmatrix} \|\mathbf{z}_1^{(t-1)} - \mathbf{o}_1^{(t)}\|_2 & \|\mathbf{z}_1^{(t-1)} - \mathbf{o}_2^{(t)}\|_2 & \ldots & \|\mathbf{z}_1^{(t-1)} - \mathbf{o}_{m(t)}^{(t)}\|_2 & d_1 & \infty & \infty & \ldots & \infty \\ \|\mathbf{z}_2^{(t-1)} - \mathbf{o}_1^{(t)}\|_2 & \|\mathbf{z}_2^{(t-1)} - \mathbf{o}_2^{(t)}\|_2 & \ldots & \|\mathbf{z}_2^{(t-1)} - \mathbf{o}_{m(t)}^{(t)}\|_2 & \infty & d_1 & \infty & \ldots & \infty \\ \|\mathbf{z}_3^{(t-1)} - \mathbf{o}_1^{(t)}\|_2 & \|\mathbf{z}_3^{(t-1)} - \mathbf{o}_2^{(t)}\|_2 & \ldots & \|\mathbf{z}_3^{(t-1)} - \mathbf{o}_{m(t)}^{(t)}\|_2 & \infty & \infty & d_3 & \ldots & \infty \\ \ldots & \ldots & \ldots & \ldots & \infty & \infty & \infty & \ldots & \infty \\ \|\mathbf{z}_n^{(t-1)} - \mathbf{o}_1^{(t)}\|_2 & \|\mathbf{z}_n^{(t-1)} - \mathbf{o}_2^{(t)}\|_2 & \ldots & \|\mathbf{z}_n^{(t-1)} - \mathbf{o}_{m(t)}^{(t)}\|_2 & \infty & \infty & \infty & \ldots & d_n \end{bmatrix} \tag{2.8}$$

The cost matrix $\mathbf{C}$ defines the GNN filter objective, while the LAP constraints complete the optimization problem. The resulting linear program is then solvable in polynomial time [35, 39], yielding globally optimal assignments between tracks and measurements. The columns $m(t) + 1, m(t) + 2, \ldots, m(t) + n$ correspond to a track receiving no detection update are included in the one-to-one constraints.

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{n} \sum_{j=1}^{m^{(t)}} \mathbf{C}_{ij}^{(t)} x_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_{ij} = 1 \qquad j = 1, \ldots n + m^{(t)} \\
& \sum_{j=1}^{m^{(t)}} x_{ij} = 1 \qquad i = 1, 2, \ldots n \\
& x_{ij} \in \{0, 1\}
\end{aligned}
\tag{2.9}
$$

The GNN is the most common tool for solving the association step in MOT. The method is typically used in conjunction with physical models of object motion to a crucial step to MOT paradigms. Dynamical models of varying complexity can model first and second order liner motion, or even nonlinear trajectories. In particular, linear dynamical models for MOT are parameterized via the Kalman Filter [28]. The Kalman Filter recursively updates parameters of a linear dynamical model which is used to generate a priori state predictions $\hat{\mathbf{z}}_i^{(t)}$, $i = 1, 2, \ldots, n$ prior to each association step. The state predictions are used in the formulation of $\mathbf{C}$, replacing the previous frame states $\mathbf{z}_i^{(t-1)}$. Each Kalman Filter is updated after the association step, such that it better reflects object behavior. Benchmark MOT performance can be achieved by using neural networks for object detection and the GNN LP with dynamical motion modeling for association [8].

2.2.2.1 Multiple Hypothesis Tracking & Joint Probabilistic Data Association

The simplicity of the GNN lends itself to a multitude of applications. However, the method often falls short in complex scenarios. The GNN fails in two distinct scenarios: cluttered scenes and coalescing trajectories. Clutter describes the problem of having many false positives amid a true object. The one-to-one constraints force one detection to be

used to update the object's state. Trajectory coalescence occurs when multiple objects intersect and move in a similar direction. A frame-to-frame MOT approach may not be able to disambiguate the objects.

Joint probabilistic data association (JPDA) and the aforementioned multiple hypothesis tracking (MHT) address these problems, respectively. The two more computationally intensive solutions to the data association problem use different techniques to handle each problem but are united in using *all* solutions to the GNN LP (Eq 2.9. Murty's algorithm iteratively solves the GNN LP and adds constraints to ensure each solution is excluded from the solution set, yielding the K best solutions with complexity $\mathcal{O}(Kn^4)$ [55]. Miller, Stone, and Cox further optimized Murty's method to $\mathcal{O}(Kn^3)$ using the Jonker-Volgenant algorithm [35, 54]. Cox and Miller also proposed changes to Murty's algorithm to better suit the method for the data association problem [23].

The original implementations of JPDA and MHT both use the $n!$ solutions of the GNN LP to formulate state updates. [4, 64]. JPDA is known as a soft-assignment method, a multi-target extension of the single target probabilistic association filter. The single target approach uses a weighted linear combination of detections with an object gate to yield a state update. The method is most effective in applications with cluttered detection sets. The extension to multi-target tracking requires enumeration of the solution space as it does not simply apply n independent filters. Enumeration of all hypotheses allows for the calculation of joint probabilities to be used in the multi-target state update without using detections more than once each [4]. On the other hand, MHT as first proposed requires the enumeration of hypotheses which are propagated to future frames in which detections are further associated with each to find the most likely hypothesis of the original

set. The recursive process constitutes a deferred decision paradigm in which challenging track associations are only made when information from future detections can disambiguate the decisions.

Murty's algorithm has proven integral to building approximations to both JPDA and MHT. For example, in JPDA, the K-best solutions are used to approximate the full joint assignment probabilities [65]. Then, only the most likely hypotheses can be generated, propagated, and maintained across future detection sets [10, 22]. The method allows for what were once worst-case exponentially scaling algorithms to be approximated in polynomial time.

In particular, MHT can be expressed as a multidimensional assignment problem (MAP). The resulting optimization problem can then be solved by branch-and-bound approaches [17, 30].

Presented contributions focus on the data association step of MOT in application to nuclei tracking in embryonic *C. elegans*. The next chapter analyzes the association objective as a forum to discuss dependencies between objects. The second leading contribution expands the MHT paradigm to model interdependencies in the association objective within the multi-frame context. The hypergraphical MAP solution is approximated from a leading MHT solution archetype.

Chapter 3: Exact Hypergraph Matching for Posture Identification in Embryonic *C. elegans*

3.1 Introduction

Point-set matching describes the task of finding an alignment between two sets of points. The problem appears in computer vision applications such as point-set registration [45], object recognition [6], and multiple object tracking [87]. Each set of points can be modeled via a hypergraph. User-defined attributes characterize the vertices and hyperedges, such as lengths of chords connecting points, or angles between triplets of points, respectively. Specified attributes give insight to observable relationships between vertices and allow for structural analyses of hypergraphs. Hypergraph matching is the optimization problem defined by the search for a correspondence of vertices between a pair of attributed hypergraphs. The permutation matrix space Π is updated to the *assignment matrix* space to account for situations in which hypergraphs are of varying sizes.

$$\mathcal{X} = \{X \in \{0,1\}^{n_1 \times n_2} : \forall j, \sum_{i=1}^{n_1} x_{ij} \leq 1, \forall i \sum_{j=1}^{n_2} x_{ij} = 1\} \tag{3.1}$$

The space $\mathcal{X}$ (Eq 3.1) comprises assignment matrices which each describe a one-to-one alignment between vertices of the two hypergraphs. Hypergraph matching then concerns

finding an optimal vertex correspondence between pairs of attributed hypergraphs. The number of vertices aligned by the most comprehensive hyperedge defines the degree of a hypergraph. Maximum degree hypergraphs with hyperedges composed of all n_1 vertices yield the most comprehensive point-set matching function possible. The optimization objective function captures the dissimilarity arising between the matching: $(l_1, l_2, \ldots, l_{n_1}) \mapsto (l'_1, l'_2, l'_3, \ldots, l'_{n_1})$. Then, for a given assignment matrix $X \in \mathcal{X}$, the hypergraph matching objective can be expressed using n_1 dissimilarity tensors of dimension $2, 4, \ldots, 2d, \ldots, 2n_1$, each measuring dissimilarity between degree d hyperedges, respectively. Define $\mathbf{Z}^{(d)}$ as the tensor mapping the dissimilarity for the degree d hyperedges. The hypergraph matching objective is expressed in Eq 3.2.

$$
\begin{aligned}
f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(n_1)}) = &\sum_{l_1=1}^{n_1}\sum_{l'_1=1}^{n_2} \mathbf{Z}^{(1)}_{l_1 l'_1} x_{l_1 l'_1} + \sum_{l_1=1}^{n_1}\sum_{l'_1=1}^{n_2}\sum_{l_2=l_1+1}^{n_1}\sum_{l'_2=1}^{n_2} \mathbf{Z}^{(2)}_{l_1 l'_1 l_2 l'_2} x_{l_1 l'_1} x_{l_2 l'_2} \\
&+ \sum_{l_1=1}^{n_1}\sum_{l'_1=1}^{n_2}\sum_{l_2=l_1+1}^{n_1}\sum_{l'_2=1}^{n_2}\sum_{l_3=l_2+1}^{n_1}\sum_{l'_3=1}^{n_2} \mathbf{Z}^{(3)}_{l_1 l'_1 l_2 l'_2 l_3 l'_3} x_{l_1 l'_1} x_{l_2 l'_2} x_{l_3 l'_3} + \ldots \\
&+ \sum_{l_1=1}^{n_1}\sum_{l'_1=1}^{n_2} \ldots \sum_{l_{n_1}=l_{n_1-1}+1}^{n_1}\sum_{l'_{n_1}=1}^{n_2} \mathbf{Z}^{(n_1)}_{l_1 l'_1 \ldots l_{n_1} l'_{n_1}} x_{l_1 l'_1} \ldots x_{l_{n_1} l'_{n_1}} \qquad (3.2)
\end{aligned}
$$

Hypergraph matching allows for the modeling of intricate point-set matching problems through high multiplicity assignment objective function formulations. The $\mathbf{Z}^{(d)}$ dissimilarity terms measure degree d hyperedge dissimilarity comprising d simultaneous vertex assignments. The range in assignment problem objective complexity from d=1 to d=n_1 trades off model capacity for increased computation. The traditional linear assignment problem (d=1) is solvable in polynomial time [39], but treats points between sets independently. Existing

graphical methods (d=2) and hypergraphical methods (d>2) rely on approximate searches and do not generalize to high degree formulations of Eq 3.2. EHGM (EHGM) is able to find globally optimal solutions to hypergraph matching problems of arbitrary degree, allowing for the modeling of intricate point-set matching tasks.

3.1.1 Overview of EHGM & Application to *C. elegans*

EHGM deviates from recent graph matching and hypergraph matching methodology as an exact method, guaranteeing convergence to a globally optimal solution (Chapter A.1). Heuristic hypergraph matching methods approximate the assignment matrix using the dissimilarity tensor [18, 27] whereas EHGM builds upon the seminal branch-and-bound algorithm [41]. EHGM extends the methodology to branch and prune based upon a given hypergraphical model. A k-tuple of nodes at branch m are greedily selected while another step encapsulates the full hypergraphical objective upon selection. These changes enable flexibility in altering the hypergraph matching objective, particularly in allowing for high degree hypergraphical modeling.

EHGM was applied to model *posture* in embryonic *Caenorhabditis elegans* (*C. elegans*), a small, free-living roundworm. The nematode features approximately 550 cells upon hatching, including a set of twenty *seam cells* and two associated neuroblasts. The seam cells and neuroblasts form in lateral pairs along the left and right sides of the worm, resulting in eleven pairs upon hatching [77]. The neuroblasts appear in the final hours of development, just prior to hatching. The pairs of cells are named, posterior to anterior: *T*, *V6*, *V5*, *Q* (neuroblasts), *V4*, *V3*, *V2*, *V1*, *H2*, *H1*, and *H0*. Each pair's left and right

cell is named accordingly; for example, *H1L* and *H1R* comprise the *H1* pair. Figure 3.1-A depicts center points of seam cell nuclei located in an example image volume as imaged in the eggshell (left) and straightened to reveal the bilateral symmetry in seam cell locations (right). Figure 3.1-B shows four sequential images of an embryo, five minutes between images.

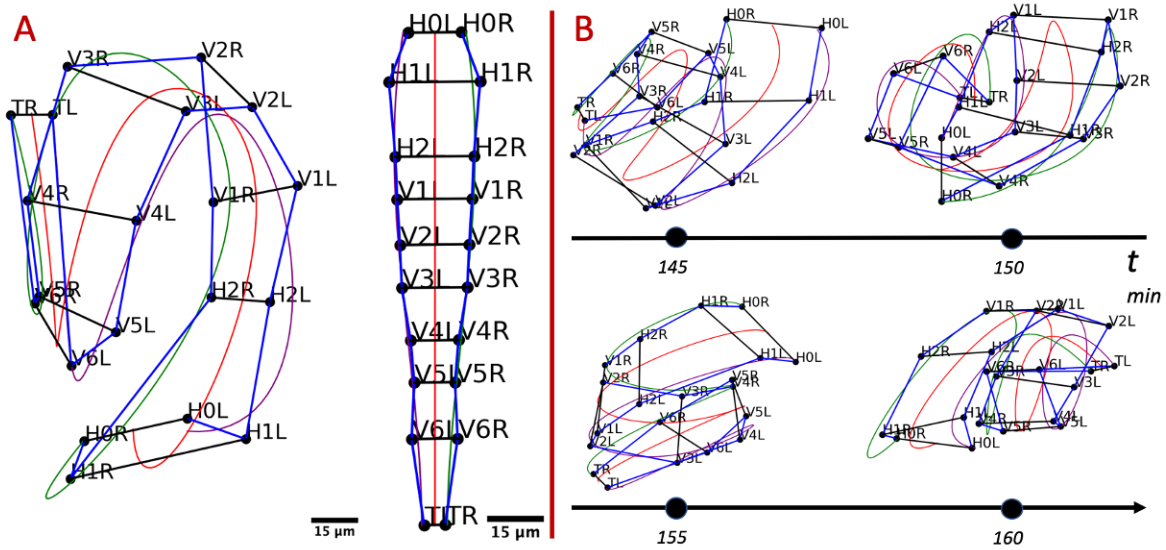

Figure 3.1: **High spatial resolution, low temporal resolution imaging necessitates posture identification.** A: Manually identified seam cell nuclei from an imaged *C. elegans* embryo. The cells form in pairs; they are labelled posterior to anterior: *T*, *V6*, ..., *H0*. The identification of all seam cells reveals the embryo's posture. Natural cubic splines through the left and right-side seam cells estimate the coiled body. The left image depicts identified nuclei connected to outline the embryonic worm. The fit splines are used to *untwist* the worm, generating the remapped straightened points in the diagram on the right. B: Labelled nuclear coordinates from a sequence of four images. The embryo repositions in the five minute intervals between images, causing failure of traditional tracking approaches.

We define *posture* as the identification of all seam cells and neuroblasts, which together reveal the shape of the coiled embryo. Posture identification allows for traditional frame-to-frame tracking of imaged cells belonging to various tissues such as the gut, nerve ring, and

bands of muscle [21]. Images are captured in five minute intervals (Figure 3.1-B) in order to achieve necessary resolution to track cells of other tissues without disturbing embryo development. Figure 3.2-A highlights muscle cell nuclei (red dots) with the identified seam cells to contextualize the embryo's positioning. The posture is used to remap the muscle cells such that traditional cell tracking approaches can be applied in the late-stage embryo (Figure 3.2-B). Figure 3.2-C depicts the cell remapping process [21]. The muscle cells are remapped according to splines fitted to the posture. The *untwisted* cell positions are then tracked frame-to-frame (Figure 3.2-D).

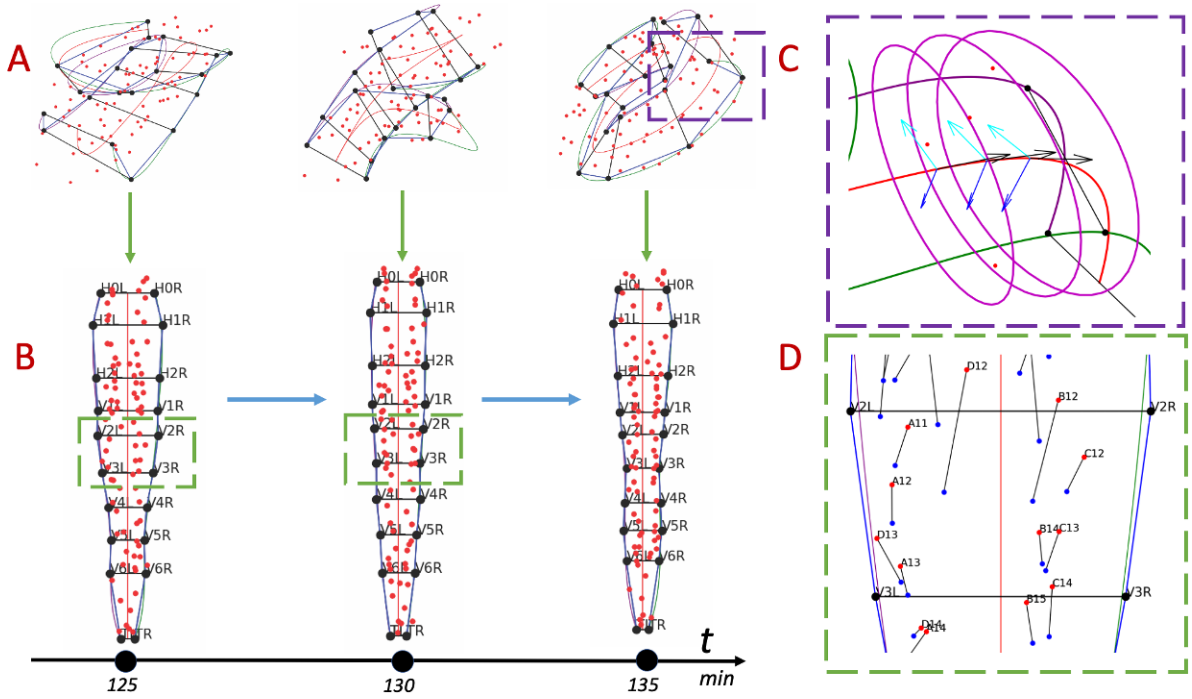

Figure 3.2: **Posture identification allows the tracking of other cells during late-stage embryogenesis.** A: Seam cell nuclei coordinates (black) and muscle nuclei coordinates (red) in a sequence of three sequential volumetric images. The untwisting process (green arrows) uses the seam cells to remap muscle coordinates to a common frame of reference. B: The remapped muscle nuclei are tracked frame-to-frame (blue arrows). C: A higher magnification view from the right coordinate plot of A. The left, right, and midpoint splines are used to create a change of basis defined by the tangent (black), normal (blue), and binormal (cyan) vectors. Ellipses are inscribed along the tangent of the midpoint spline, approximating the skin of the coiled embryo. D: A portion of the left (red) and center (blue) remapped muscle coordinates. Black lines connect the coordinates, frame-to-frame.

Current methods for posture identification rely on trained users to manually annotate the imaged nuclei using a 3D rendering tool [53]. The process takes several minutes per image volume and must be performed on approximately 100 image volumes per embryo [21]. Manual annotation strategies motivated us to develop EHGM, as established methods for point-set matching fail to adequately capture the relationships between seam cells throughout myriad twists and deformations of the developing embryo. Figure 3.3 depicts manually identified postures in the first two successive image volumes of Figure 3.1-B. Manual identification is performed in Medical Imaging, Processing, Analysis and Visualization (MIPAV), a 3D rendering program used for manual annotation [53]. Posture identification can be phrased as a MOT problem in which the seam cells are to be tracked despite five minute intervals between images.

EHGM used hypergraphical models comprising biologically driven geometric features to more accurately identify posture than established graphical methods. The limited expressive power of graphical models hindered accurate seam cell identification; graphical models accurately identified posture in 27% of samples compared to 56% using a hypergraphical model. User labelling of the posterior-most seam cell nuclei improved the success of hypergraph matching to correctly identifying all nuclei in 77% of samples. The improved accuracy in posture identification attributed to high-degree hypergraphical modeling solved via EHGM paves a path toward automatic posture identification while presenting a general framework for approaching similarly challenging point-set matching tasks.

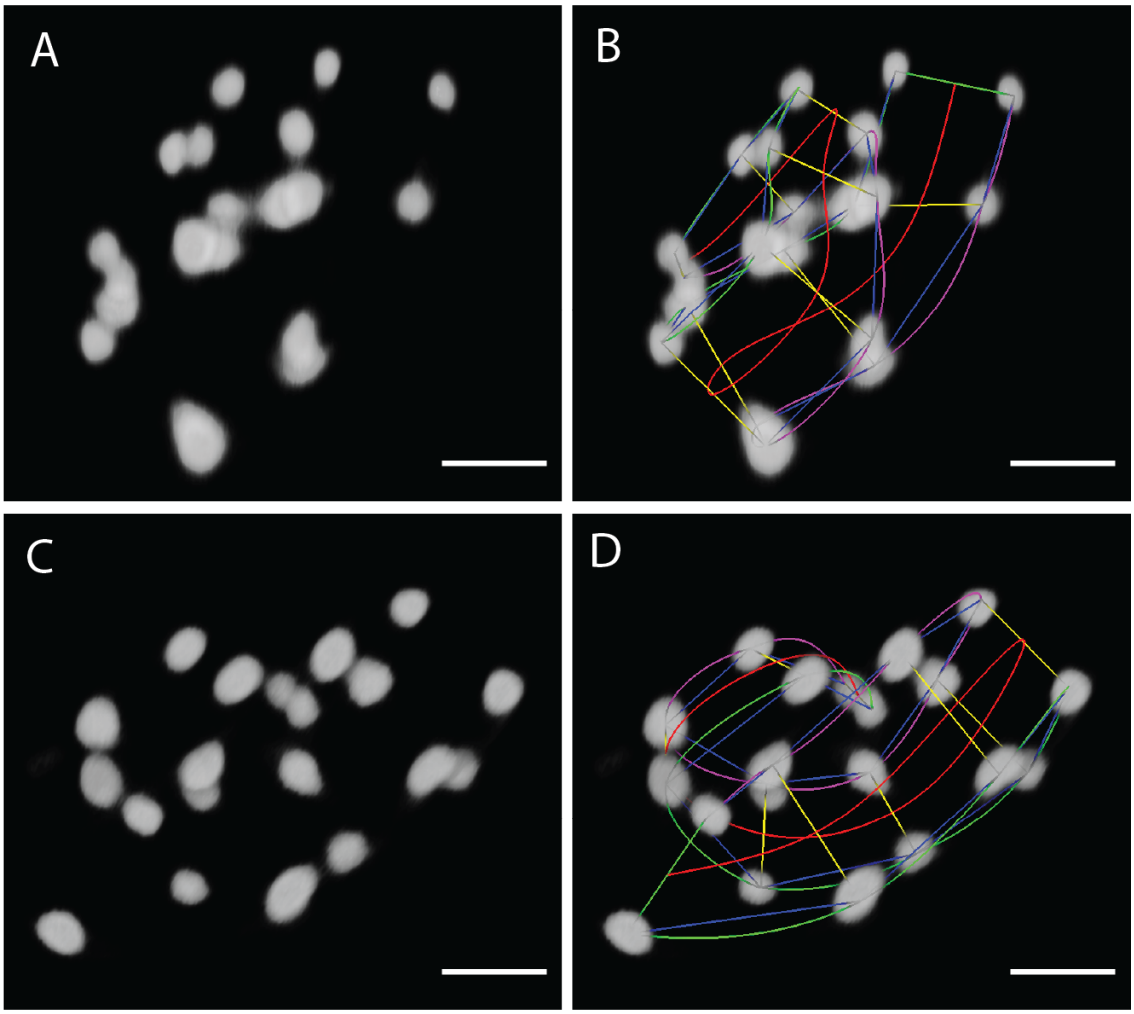

Figure 3.3: **Manual posture identification in two successive image volumes of Figure 3.1-B using MIPAV.** The 20 fluorescently imaged seam cell nuclei rendered in two successive image volumes. Scale bar: 10 μm. A & B: Seam cell nuclei appearing in two successive image volumes visualized in MIPAV. The five minute interval allows the embryo to reposition between images, yielding entirely different postures. C & D: Manual seam cell identification by trained users reveals the posture. The curved lines are cubic splines as described in Figure 3.2-C.

3.2 Results

3.2.1 Posture Identification Models

Posture was predicted via EHGM according to three models: a graphical model, denoted *Sides*, and two hypergraphical models. The two hypergraphical models, *Pairs* and

Posture, showcase EHGM as existing algorithms cannot find solutions under such high degree hypergraphs. Each of the three models incrementally use higher degree terms to describe posture. *Sides* follows the form of Eq 2.6 and leverages pairwise assignments to calculate lengths and widths of portions of the embryo. *Pairs* uses degrees four and six hyperedges to better model local regions of the embryo than is possible with graphical methods which rely on pairwise relationships. *Posture* further demonstrates the capabilities of EHGM by including a degree n_1 hyperedge to maximize context in evaluating a hypothesized posture. Geometric features such as pair-to-pair rotation angles and left-right flexion angles were developed to more accurately measure and compare posture hypotheses. The calculation of each angle or distance requires identification of multiple seam cells in tandem to calculate, necessitating the use of hyperedges.

Models in this sense can be viewed as assignment problem objectives of differing specification and complexity. An assignment objective $f : \mathcal{X} \mapsto R$ will take as input a permutation matrix subject to points given in each point-set to yield an objective function value $f(X)$. The LAP and the QAP have been used extensively to perform point-set matching in image processing due to the availability of computationally efficient algorithms yielding satisfactory solutions for simpler tasks.

Consider a model M that takes in a known *correct* permutation matrix $\bar{X} \in \diamond$ given a point-set $\mathbf{X}$ and outputs cost $\bar{C}$. Suppose also the globally optimizer $X^* \in \Pi$ (given the same point-set input $\mathbf{X}$) outputs the globally optimal cost $M(X^*) = C^*$. Let $\mathbf{X}$ denote a point-set and $\bar{X}$ an ordering. One relevant question is how to design M such that for a given set of pairs $\{(\mathbf{X}_1, \bar{X}_1), (\mathbf{X}_2, \bar{X}_2), \ldots, (\mathbf{X}_N, \bar{X}_N)\}$ one can find:

$$M^* = \underset{M}{\operatorname{argmin}} \frac{1}{N} \sum_{L=1}^{N} \|\bar{C}_L - C_L^*\|_2$$

Figure 3.4 demonstrates four types of models applied to perform posture identification on the first two sampled images in Figure 3.1-B. Linear models (Figure 3.4-A & Figure 3.4-B) are ill-equipped to identify posture due to the repositioning of the embryo between successive images, so linear models are not evaluated on sampled data. The graphical model *Sides* (Figure 3.4-C & Figure 3.4-D) associates local seam cells via edges (purple). Edge-wise features such as lengths and widths vary if the embryo coils tightly, but are otherwise approximately static frame-to-frame. However, the similarity in these measurements throughout the embryo yields a model incapable of differentiating portions of the embryo. Hypergraphical models *Pairs* (Figure 3.4-E & Figure 3.4-F) and *Posture* (Figure 3.4-G & Figure 3.4-H) use aforementioned hyperedges to more strongly characterize embryonic posture.

3.2.2 Posture Identification Accuracy

Annotators curated a dataset of seam cell nuclei center coordinates from 16 imaged embryos. Each imaged embryo yielded approximately 80 image volumes for a total of N=1264 labelled seam cell nuclei coordinate sets. Homogeneity in *C. elegans* embryo development allowed use of samples spanning multiple embryos to fit models via a leave-one-out approach (A.1: *Model Fitting*, A.1: *Posture Modeling*). EHGM allows for known correspondences, henceforth referred to as *seeds*, to be given as input prior to search initialization. The algorithm was evaluated both in a traditional point-set matching scenario given no *a priori* information, and in a series of seeded simulations. Seeded trials assumed

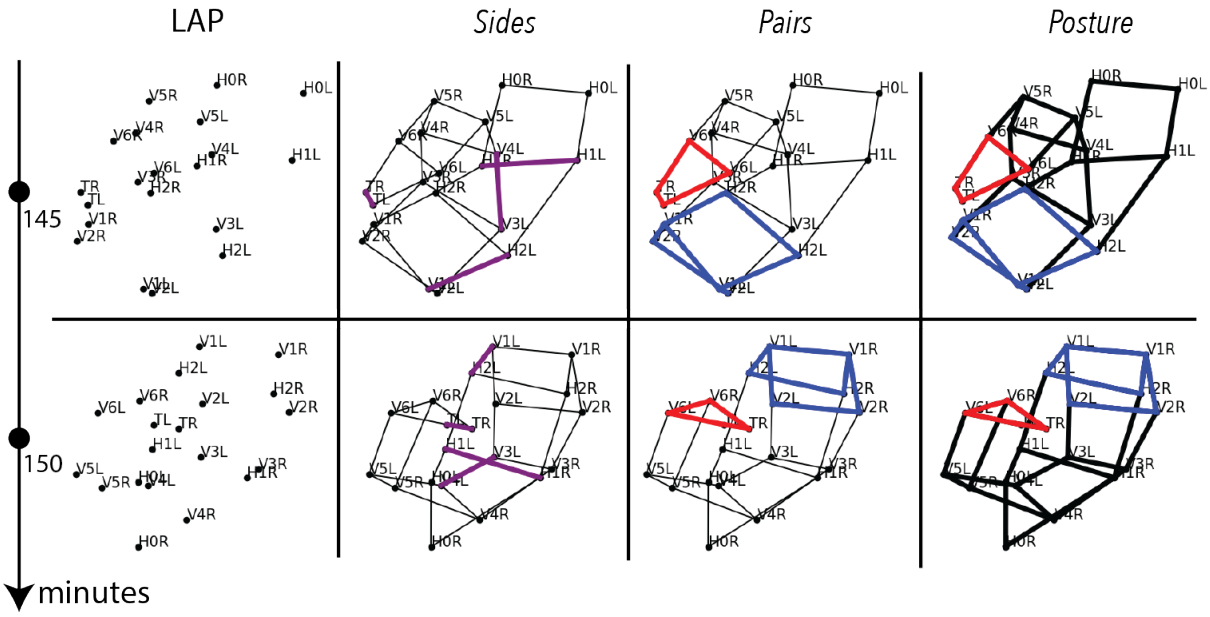

Figure 3.4: **Posture identification applied to the two successive images in Figure 3.3 according to a series of increasingly intricate models.** The embryo repositions between images. A & B: Linear models (LAP) cannot quantify relationships between seam cells; posture identification is impossible without context of neighboring cell identities. C & D: A graphical model (*Sides*) specifies edges (purple) between pairs of seam cell nuclei. Edge lengths are relatively static frame-to-frame, but the similarity of edge lengths throughout the embryo causes the edges to have a weak signal in identifying seam cells. E & F: The *Pairs* model uses degrees four (red) and six (blue) hyperedges to model a greater local context than is possible in a graphical model. G & H: The *Posture* model extends the *Pairs* model to use a degree n_1 (black) hyperedge to evaluate all seam cell assignments jointly.

incrementally more pairs given sequentially from the tail pair, *T*, to the fourth pair, *V4* (or *Q* for n_1=22 samples). *KerGM* [96], a leading algorithm for heuristic graph matching, was applied to posture identification. The algorithm used the same connectivity matrix as *Sides*, but processed results frame-to-frame serially, relying on the correct posture identification at the prior image as input to search.

EHGM is able to store complete assignments encountered during the search as it compares against the current solution at the final branch. This allowed for an analysis of the similarity between cost minimizing posture hypotheses and progressively higher cost

solutions encountered during search. The top x accuracy describes the percentage of all N samples in which EHGM returned the correct posture in the x lowest cost solutions; i.e. the top 1 accuracy describes the percentage of samples in which the correct posture was returned as the cost minimizing posture, and the top 3 accuracy is the percentage of samples in which the correct assignment was among 3 lowest cost posture hypotheses returned by the search. Top x accuracies are reported alongside the median runtime and the median cost ratio. The cost ratio is defined as the ratio of the correct posture's objective to the cost minimizing posture's objective. A cost ratio greater than one implies the objective of the hypothesized posture is lower than that of the correct posture, suggesting the model is not aptly characterizing posture as an incorrect posture hypothesis was preferred by the model.

Table 3.1 shows the percentage of all N samples in which the correct posture (correct identification of *all* seam cells) was returned as the minimizer according to *KerGM* and each of the models solved via EHGM: *Sides*, *Pairs*, and *Posture*. *KerGM* identified 27% of sampled postures correctly, outperforming *Sides* (10%). *Pairs* and *Posture* more effectively identified posture with 52% and 56% top 1 accuracies, respectively. Both hypergraphical models also reported a median cost ratio of 1.00, compared to 1.28 of *Sides*, suggesting the hypergraphical representations of coiled posture provided enhanced discriminatory power across samples. The hypergraphical models demonstrated small trade-offs between accuracy and runtime. The *Posture* model's n_1 degree hypergraphical features improved accuracy over *Pairs*, 56% to 52%, in exchange for longer median runtime, 60 minutes to 43 minutes. Differences between the top 1 and top 3 accuracies reflect the challenge in posture identification. The optimums under the *Pairs* and *Posture* models were often

similar to those of similar posture hypotheses. Notably, the *Posture* model returned the correct posture in the top 3 hypotheses in approximately 67% of samples, an approximate 20% increase in relative accuracy over the top 1 percentage, 56%.

	Top 1 (%)	Top 2 (%)	Top 3 (%)	Top 5 (%)	R (minutes)	CR
KerGM	27	27	27	27	.01	
Sides	10	14	15	16	5.97	1.28
Pairs	52	60	63	65	43.22	1.00
Posture	56	65	67	68	60.35	1.00

Table 3.1: **Hypergraphical model *Posture* achieves highest accuracy.** Posture identification accuracies across all N=1264 samples. *KerGM* is compared to proposed models. The first columns list the top x accuracy as a percentage of samples. The column titled R shows the median runtime of each model in minutes. *CR* reports the median cost ratio, defined as the ratio of the correct posture cost to the returned posture cost.

Posture identification results were stratified by the presence of the Q neuroblasts; 875 of the 1264 samples contain only the seam cells while the remaining 389 samples are mature enough to have the Q neuroblasts. Table 3.2 depicts the findings presented in Table 3.1 split by Q neuroblast presence. *KerGM* and all models solved via EHGM achieved a higher accuracy on Q samples. Notably, the *Posture* model's top 3 accuracy is higher on the Q samples (82%) than the pre-Q samples (60%). The extra pair of coordinates provided substantial context, further defining the coiled shape and helping to penalize incorrect postures.

The large increases from top 1 accuracy to top 5 accuracy for the *Pairs* and *Posture* models prompted a further investigation into the leading hypotheses across samples. A fourth hypergraphical model, *Full*, comprising only the degree n_1 hyperedge features was conceptualized as a powerful model to compare a small set of hypotheses. The hypotheses for each sample were ranked according to the *Full* model. The final comparison improved

	Top 1 (%)	Top 2 (%)	Top 3 (%)	Top 5 (%)	R (minutes)	CR
KerGM	25	25	25	25	.01	
Sides	7	10	11	12	4.81	1.36
Pairs	44	51	55	57	34.25	1.04
Posture	48	57	60	61	51.12	1.00
	Top 1 (%)	Top 2 (%)	Top 3 (%)	Top 5 (%)	R (minutes)	CR
KerGM	35	35	35	35	.01	
Sides	19	25	26	26	9.66	1.16
Pairs	71	80	82	82	56.58	1.00
Posture	72	81	82	83	72.60	1.00

Table 3.2: **Hypergraphical models leverage Q neuroblasts to identify posture.** The samples are split according to the absence (top) or presence (bottom) of the Q neuroblasts, which form in the last two hours of development. There are 875 n_1=20 cell samples and 389 n_1=22 Q samples. Reported methods more accurately identify embryonic posture in the Q samples, suggesting the increased continuity along the body of the embryo allows for more consistent posture identification.

performance for the simplest model, *Sides*; the top 1 accuracies improved from 7% to 12% on the pre-Q data and from 19% to 24% on Q samples. However, the ranking reduced performance on the hypergraphical models *Sides* (pre-Q: 44% to 40%, Q: 71% to 58%) and *Posture* (pre-Q: 48% to 39%, Q: 72% to 54%).

Seeded experiments specifying nuclear identities provided *a priori* information starting with the tail pair, and incrementally identified more pairs in the posterior region. Each experiment was given five minutes of maximum runtime; a semi-automated solution requiring more runtime was deemed infeasible. Top 1 and top 3 accuracy percentages are reported by EHGM models and number of seeded pairs in Table 3.3. Seeding yielded decreasing marginal improvements to accuracy and runtime. Figure 3.5 depicts top 1 accuracies and median runtimes across seeded experiments for the *Pairs* and *Posture* models split by Q pair labelling. Particularly, seeding the first two pairs, T and *V6*, greatly reduces the median runtime while also netting the largest gains in top 1 accuracy, partially attributable

to EHGM converging in the given timeframe.

	Top 1 (%)					Top 3 (%)				
	None	T	T-V6	T-V5	T-V4	None	T	T-V6	T-V5	T-V4
Sides	9	10	22	29	37	13	15	27	35	43
Pairs	34	49	72	79	84	38	54	77	83	87
Posture	25	36	68	79	84	27	39	73	84	87

Table 3.3: **Seeding posterior pair identities promotes accurate posture identification and reduces runtime.** Top 1 and top 3 seeded posture identification accuracies across all samples. All trials had a five-minute maximum runtime. The rows again correspond to each model. Columns specify which pairs were given as seeds prior to search. The *None* columns recreate the original no information task. The subsequent columns specify which pairs are correctly identified prior to search.

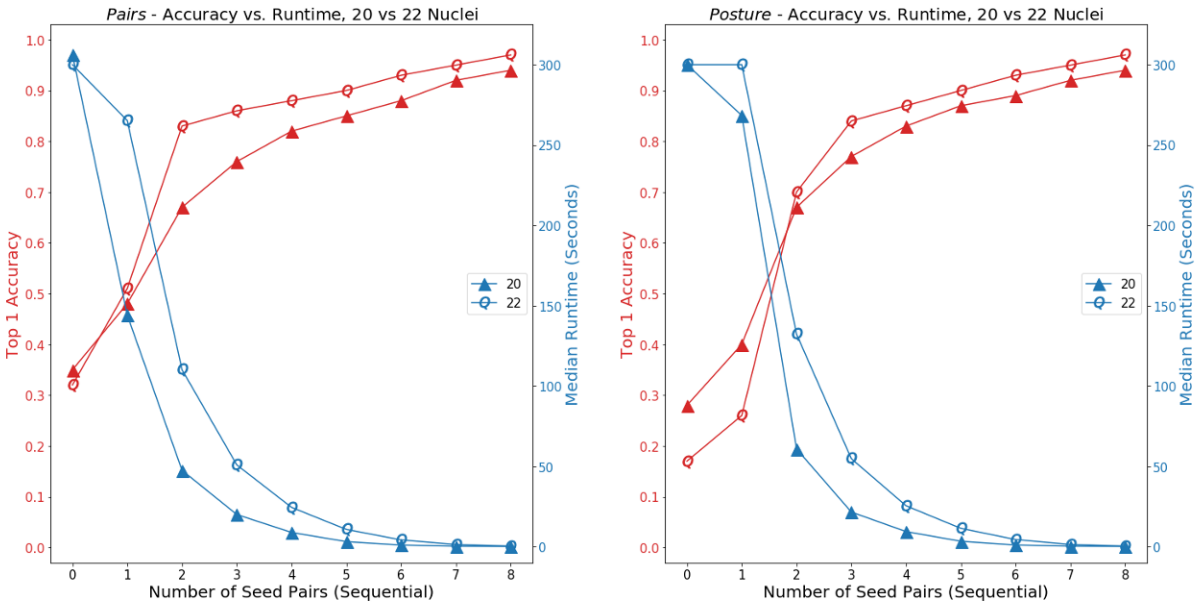

Figure 3.5: **Evaluating the *Pairs* and *Posture* models as seam cell identities were seeded.** The *Pairs* and *Posture* models top 1 accuracies and median runtimes by Q pair labelling. Posterior pair seeding drastically improved top 1 accuracy and reduced runtime when applying both models. Q pair samples required more runtime (n_1=22 as opposed to n_1=20), but the added context improved posture identification accuracy. The majority of samples converged within 5 minutes when seeded with the T and $V6$ pairs of nuclei.

3.3 Discussion

We have presented EHGM as a dynamic and effective tool for intricate point-set matching tasks. The hypergraph matching algorithm provides a method in which to gauge the efficacy of modeling point correspondences in conservatively-sized problems; problems featuring larger numbers of points likely contain the context required to match adequately via lower degree models. For example, postures in samples containing Q nuclei were more accurately identified across models, but the largest marginal gain in accuracy came from *Sides* (d=2) to *Pairs* (d=4,6). The results suggest that added context throughout the embryo would further improve posture identification accuracy, reducing the reliance on higher degree (and thus more computationally expensive) hypergraphical objective function formulations. EHGM specifically addresses a gap in literature concerning challenging point-set matching applications in which domain-specific features lead to rigorously testable models. Seeding allows a wider range of problems to be approached, and mitigates the computational expense of the algorithm for scenarios featuring larger point-sets.

Posture identification in embryonic *C. elegans* is a challenging problem benefiting from high degree hypergraphical modeling. EHGM equipped with biologically inspired hypergraphical models led to substantial improvement in posture identification. The top 1 accuracy doubled from 27% with a graphical model to 56% via the *Posture* model (Table 3.1). The top 3 accuracy rate improved to 67%, highlighting the challenge in precisely specifying the coiled embryo due to the similarity of competing posture hypotheses. The presence of Q neuroblasts further contributed to accurate posture identification. The added context empowered the *Posture* model to identify the correct posture in 82% of Q

samples (Table 3.2.

The top x percentage accuracy metric reflects the need to correctly identify *all* seam cells in order to recover the underlying posture, but does not distinguish between hypotheses that are incorrect due to one cell identity swap or a more systemic modeling inadequacy. A qualitative analysis highlighted a few themes among incorrectly predicted postures. The foremost errors concern the tail pair cells, *TL* and *TR*; spurious identifications occurred when the tail pair coiled against another the body of the embryo, causing one tail cell identity to be interchanged with a cell of a nearby body pair. The variance of feature measurements in the posterior region resulted in similar costs for postures with minor differences about the posterior region. Further evaluation on posture-wide hypergraphical model (*Full*) on top posture hypotheses proved to be ineffective.

Pair seeding allows for the strengths of EHGM to compensate for the most challenging aspect of posture identification. The posterior region of the embryonic worm is especially flexible and contributes to the majority of reported errors. Feature engineering stands to create hypergraphical models more capable of reliable posture identification, particularly in contextualizing the posterior region. The method and application outline a protocol for challenging point-set matching tasks.

EHGM enables the modeling and approach of previously intractable point-set matching tasks. The flexibility of the paradigm in designing models and using data to estimate parameters in an offline fashion adds further enables applicability. Our presented "template" matching approach stands in contrast to traditional methods for modeling transformations in which identified points are assigned to a set of unidentified points arising in a subsequent frame [27, 95, 97]. EHGM can be applied in the frame-to-frame manner as well; the

model fitting procedure (Chapter 3.5.3) simply requires using frame-to-frame differences to estimate parameters of each hyperedge's distribution.

3.4 Future Work

Automated posture identification is infeasible with current approaches. EHGM with proposed models did not reliably identify posture. The *Posture* model's accuracy increased 11% when considering the top 3 solutions instead of the top solution. The combinatorially large space ($\approx 10^{18}$-10^{21}) being accurately reduced to the top 1 or top 3 is evidence of the *Posture* model accurately characterizing the embryo. As noted, many errors comprised just the tail pair. Moreover, the difference between the optimal value and correct posture's costs were often similar (median CR=1). In summary, our proposed model does not perform adequately despite accurately contextualizing seam cell interdependencies. Theoretically there could exist a model which does reliably predict posture. Features used *Posture* were derived from cues experts used in the manual annotation process. An iterative model fitting paradigm akin to that of [14] could better tune existing models as well. Further simulations in the traditional frame-to-frame matching paradigm could also be pursued.

3.5 Methods

3.5.1 EHGM

EHGM extends the branch-and-bound paradigm to exactly solve hypergraph matching. The algorithm performs the search in the permutation space $\mathcal{X}$ subject to a given branch size k which specifies the number of vertices assigned at each branch. A size n_1 hypergraph

will require $M := \frac{n_1}{k}$ branch steps, where branch m concerns the assignment of vertices $((m-1)k+1, (m-1)k+2, \ldots, mk)$; vertices $1, 2, \ldots, mk$ have been assigned upon completion of the m^{th} branch. The set $\mathbf{P}$ contains all possible permutations of the indices of the unordered point-set, $|\mathbf{P}| = \frac{n_2!}{(n_2-k)!}$. $\mathbf{P}$ is incrementally subset into queues $\mathbf{Q}_m \subseteq \mathbf{P}$ at branches $m = 1, 2, \ldots, M$ at each branching. The queue $\mathbf{Q}_m$ is subset according to both a pruning rule which eliminates permutations leading to a suboptimal solution as well as the one-to-one constraints of $\mathcal{X}$. The search converges to a global optimum upon the implicit enumeration of $\mathbf{Q}_1 = \mathbf{P}$.

The objective function f is further stratified according to the branch size k. Lower degree ($d \leq 2k$) hyperedge dissimilarity tensors are computed prior to search. Branches comprising k-tuples of vertices are partially assigned in a greedy manner according these lower degree hyperedge dissimilarities via the selection rule H. Later branches accrue higher degree ($d > 2k$) hyperedge dissimilarities which are calculated at time of branching; the intent of the method is to rely on lower degree terms to steer the search towards an optimum in effort to minimize the number of branches explored. The aggregation rule I accrues higher degree hyperedge dissimilarity terms upon branching, further guiding the pruning step and ensuring the complete specification of the objective f.

The branching and selection rules are designed to reduce computation performed throughout the search. A partial assignment at branch m: $\mathbf{K}_m = (l'_{(m-1)k+1}, l'_{(m-1)k+2}, \ldots, l'_{mk}) \in \mathbf{Q}_m$ is selected via precomputed lower degree hyperedge dissimilarity tensors $\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(2k)}$. A larger branch size k results in a selection rule with larger scope of the optimization landscape, better equipped to place optimal branches earlier in each queue $\mathbf{Q}_m$ at time of branching. However, computing the lower degree dissimilarity tensors prior to search can

be prohibitively expensive for larger point-sets.

3.5.2 Selection & Aggregation

The first branch permutation $\mathbf{K}_1 = (l'_1, l'_2, \ldots, l'_k) \in \mathbf{Q}_1 = \mathbf{P}$ assigns vertices $(l_1, l_2, \ldots, l_k)$ to points $(l'_1, l'_2, \ldots, l'_k)$ according to the initial branch selection rule H_1. Eq 3.3 defines a cost given dissimilarity tensors $\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots \mathbf{Z}^{(k)}$ according to a permutation $\mathbf{K}_1$. The k pairs of constraints given by the branch m and permutation of point indices $\mathbf{K}_m$: $\{(l_1, l'_1), \ldots, (l_k, l'_k)\}$ enables a simplification in the objective formulation.

$$H_1(\mathbf{K}_1|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, ..., \mathbf{Z}^{(k)}) :=$$
$$\sum_{i_1=1}^{k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{k} \sum_{i_2=i_1+1}^{k} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + ... + \sum_{i_1=1}^{k} \sum_{i_2=i_1+1}^{k} \cdots \sum_{i_k=i_{k-1}+1}^{k} \mathbf{Z}^{(k)}_{l_1 l'_{i_1} l_{i_2} l'_{i_2} \ldots l_{i_k} l'_{i_k}} \quad (3.3)$$

Subsequent branches $m = 2, 3, \ldots M$ then use the general selection rule H_m to order the permutations of the m^{th} branch: $\mathbf{K}_m = (l'_{(m-1)k+1}, l'_{(m-1)k+2}, \ldots l'_{mk}) \in \mathbf{Q}_m$. Branch $\mathbf{K}_m$ incurs a selection rule cost H_m according to Eq 3.4 comprising lower degree hyperedge dissimilarities for assignments both within branch m and the assignments between branches $1, 2, \ldots, m-1$ and branch m. The partial assignment constraints $\mathbf{K}_m$ allow further simplification of notation; the reversed order of summation indices satisfies the criteria that only hyperedge dissimilarities pertaining to branch m assignments are considered via H_m.

$$H_m(\mathbf{K}_m|\mathbf{K}_1,...,\mathbf{K}_{m-1},\mathbf{Z}^{(1)},...,\mathbf{Z}^{(2k)}) :=$$
$$\sum_{i_1=(m-1)k+1}^{mk} \mathbf{Z}^{(1)}_{l_{i_1}l'_{i_1}} + \sum_{i_2=(m-1)k+1}^{mk}\sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}}$$
$$+ \sum_{i_3=(m-1)k+1}^{mk}\sum_{i_2=1}^{i_3-1}\sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(3)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}l_{i_3}l'_{i_3}} + ... + \sum_{i_{2k}=(m-1)k+1}^{mk}\sum_{i_{2k-1}=1}^{i_{2k}-1}\cdots\sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2k)}_{l_{i_1}l'_{i_1}...l_{i_{2k}}l'_{i_{2k}}} \quad (3.4)$$

The greedy selection rule orders queues $\mathbf{Q}_m$, but does not account for higher degree ($2k < d \leq n_1$) hyperedge dissimilarities. Precomputing higher degree dissimilarity tensors can be both computationally expensive, and inefficient as ideally only a small percentage of combinations are queried throughout the search. The aggregation rule I_m, $m = 3, 4, \ldots, M$ measures the dissimilarity attributable to higher degree ($2k < d \leq mk$) hyperedges accessible due to branch m partial assignments. The aggregation rule updates the cost of branch $\mathbf{K}_m$ assignments, further informing the pruning step to subset the next queue $\mathbf{Q}_{m+1}$. The greedy selection rule H_m in tandem with the aggregation rule I_m aim to minimize the total computation performed in finding an optimum. The definition I_m follows from the general selection rule H_m, but is applied to the higher degree hyperedge dissimilarities. The aggregation rule I_m (Eq 3.5) can be expressed as the degree d dissimilarities calculable upon assignments of branch m assignments for degrees $2k < d \leq mk$.

$$I_m(\mathbf{K}_m|\mathbf{K}_1,\mathbf{K}_2,\ldots,\mathbf{K}_{m-1},\mathbf{Z}^{(2k+1)},\ldots,\mathbf{Z}^{(mk)}) :=$$
$$\sum_{d=2k+1}^{mk}\sum_{i_d=(m-1)k+1}^{mk}\sum_{i_{d-1}=1}^{i_d-1}...\sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1}l'_{i_1}...l_{i_d}l'_{i_d}} \quad (3.5)$$

The m^{th} branch allows for hyperedge dissimilarities up to degree mk concerning the first mk assignments. The M^{th} branch yields a complete assignment, allowing the evaluation of maximum degree n_1 hyperedge dissimilarities. The partitioning and further regrouping of each H_m and I_m as defined fully accounts for the objective f while allowing efficient computation during the search (A.1:*Hypergraphical Objective Decomposition*, A.1:*Convergence of EHGM*).

3.5.3 Model Fitting

Hypergraphical models comprising handcrafted features are used to contextualize relationships among a set of points. Features can be engineered and analyzed in context of point-set matching just as in traditional supervised learning tasks. Annotated data can be used to fit these hypergraphical models via measuring average feature measurements as well as the variation of a feature and covariances between features with the goal of further describing relationships between points. The hypergraphical model then describes relationships between points weighted by observation of a training set.

Features can be measured and subsequent matching performed in two ways. The first way is most commonly seen in point-set matching tasks. Here, an established hypergraph is observed at time *t-1*; points with their identities and a hypergraphical representation with features are available. A second point-set arises at time t with the goal of finding an optimal alignment of such points to maintain similarity to the reference hypergraph at time *t-1*. In this case, the established feature means are defined by the measurements of each feature in the reference hypergraph. Covariances are calculated from the frame-to-frame

differences in feature measurements across the training set. These statistics contextualize the difference between an assignment of the new point-set to the reference.

The second method measures feature means directly to aggregate feature measurements as opposed to frame-to-frame differences. This template method allows for observing features which may vary highly frame-to-frame, but follow a Gaussian distribution when viewed across samples. The distinction comes into practice when measuring feature means and covariances.

Features are expressed as attributes over hyperedge multiplicities $d = 1, 2, \ldots, n_1$. Assume there are M_d degree d hyperedges each with m_{M_d} features. The hyperedges $a = 1, 2, \ldots, M_d$ and hyperedge features $g_{a,s}^{(d)}$, $s = 1, \ldots, m_{M_d}$ are given as input. Each feature $g_{a,s}^{(d)}$ assumes a Gaussian distribution, features bundled together are modeled as a multivariate Gaussian distribution. Measurements from the data are used to obtain estimates of the parameters of the Gaussian distributions: $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. The reference hypergraph (either the previous frame hypergraph or template hypergraph) and hypergraph built during search are compared using feature measurements weighted by observed covariation in training data. The measurements calculated given assignments in the second point-set add up to a total dissimilarity cost.

The dissimilarity costs penalize the differences between a hypothesized assignment's feature measurements and the reference hypergraph feature measurements. The dissimilarity tensor $\mathbf{Z}^{(d)}$ measures the degree d hyperedge dissimilarities arising from a permutation given the M_d hyperedges each with the m_{M_d} features. Covariances between hyperedge features can be estimated *within* a hyperedge and *between* hyperedges. For example, consider hyperedges **a** (indexed by a) with n_a=2 features and **a'** (indexed by a') with $n_{a'}$=4

features, where hyperedges are of degree d. Hyperedge **a** feature covariance (between the two features) can be estimated as well as the covariances between hyperedge **a'** features (6 covariances). Additionally, the covariances between hyperedge a and hyperedge **a'** features (28 covariances) can be calculated. This may be useful for adjacent hyperedges. However, if two degree d hyperedge feature covariances are estimated, each with d distinct assignments, then the resulting covariance is of degree $2d$ as $2d$ simultaneous assignments are required to compute the dissimilarity.

For example, consider the a^{th} degree d hyperedge and its m_a features: $\mathbf{g}_a^{(d)} = [g_{a,1}^{(d)}, g_{a,2}^{(d)}, \ldots, g_{a,m_a}^{(d)}]$. The expected values: $\bar{\mathbf{g}}_a^{(d)}$ are calculated in aggregate from training data:

$$\bar{\mathbf{g}}_{a,s}^{(d)} = \frac{1}{N} \sum_{L=1}^{N} g_{a,s}^{(d)}(X_L, \mathbf{X}_L) \tag{3.6}$$

where X_L and $\mathbf{X}_L$ are the correct permutation and observed point-set, respectively, for sample $L = 1, 2, \ldots, N$. The estimated covariance matrix per the hyperedge a of the degree d hyperedges across m_a features is expressed:

$$\hat{\sigma}_{q,r}^{(d)} = \frac{1}{N} \sum_{L=1}^{N} (g_{a,q}^{(d)}(X_L, \mathbf{X}_L) - \bar{g}_{a,q}^{(d)})(g_{a,r}^{(d)}(X_L, \mathbf{X}_L) - \bar{g}_{a,r}^{(d)}) \tag{3.7}$$

$$\hat{\mathbf{\Sigma}}_a^{(d)} = \begin{bmatrix} \hat{\sigma}_{1,1}^{(d)} & \hat{\sigma}_{1,2}^{(d)} & \ldots & \hat{\sigma}_{1,m_a}^{(d)} \\ \hat{\sigma}_{2,1}^{(d)} & \hat{\sigma}_{2,2}^{(d)} & \ldots & \hat{\sigma}_{2,m_a}^{(d)} \\ \ldots & \ldots & \ldots & \ldots \\ \hat{\sigma}_{m_a,1}^{(d)} & \hat{\sigma}_{m_a,2}^{(d)} & \ldots & \hat{\sigma}_{m_a,m_a}^{(d)} \end{bmatrix} \tag{3.8}$$

The dissimilarity tensors $\mathbf{Z}^{(d)} \in R^{\underbrace{n_1 \times n_2, \ldots, n_1 \times n_2}_{2d}}$ use both sets of estimates to relay costs. The Mahalanobis distance is used to describe the scaled distance between the observed attributed hyperedge to an estimated feature description. A full assignment: $(l_1, \ldots, l_{n_1}) \mapsto (l'_1, \ldots, l'_{n_1})$ invokes costs of each hyperedge degree $d = 1, 2, \ldots, n_1$. Consider just the degree d hyperedge dissimilarities. The M_d hyperedges $a = 1, 2, \ldots, M_d$ each with m_a features comprise the degree d hyperedge dissimilarities.

Then, the degree d hyperedge dissimilarity is expressed as a summation over hyperedge dissimilarities $a = 1, 2, \ldots M_d$, each with its $s = 1, 2, \ldots, m_a$ features. Each degree d hyperedge a is defined by a tuple of d vertices: $\mathbf{a} = (a_1, a_2, \ldots, a_d)$. The accompanying assignment $\mathbf{a}' = (a'_1, a'_2, \ldots, a'_d)$ define the correspondence for referencing the cost tensor $\mathbf{Z}^{(d)}$:

$$\mathbf{Z}^{(d)}_{l_{a_1} l_{a'_1} l_{a_2} l_{a'_2} \ldots l_{a_d} l_{a'_d}} = (\mathbf{g}_a^{(d)} - \bar{\mathbf{g}}_a^{(d)})'(\hat{\mathbf{\Sigma}}_a^{(d)})^{-1}(\mathbf{g}_a^{(d)} - \bar{\mathbf{g}}_a^{(d)}) \tag{3.9}$$

Finally, the full degree d dissimilarity score can be found by considering each of the $a = 1, 2, \ldots, M_d$ hyperedges and their dissimilarities accordingly. The $\mathbf{Z}^{(d)}$ subscripts will depend on the hyperedge a and accompanying match a'. Denote $\mathbf{a} = (a_1, a_2, \ldots, a_d)$ and $\mathbf{a}' = (a'_1, a'_2, \ldots, a'_d)$ each hyperedge a's assignment from the full assignment $(l_1, \ldots, l_{n_1}) \mapsto (l'_1, \ldots, l'_{n_1})$. The complete degree d hypergraphical cost arising from the full assignment is the summation of each hyperedge a's matching cost:

$$\sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \ldots \sum_{l_d=l_{d-1}+1}^{n_1} \sum_{l'_d=1}^{n_2} \mathbf{Z}^{(d)}_{l_1 l'_1 \ldots l_d l'_d} = \sum_{\mathbf{a}} \mathbf{Z}^{(d)}_{l_{a_1} l_{a'_1} l_{a_2} l_{a'_2} \ldots l_{a_d} l_{a'_d}}. \tag{3.10}$$

The process is applied to all $d = 1, 2, \ldots, n_1$ hyperedges to to build all cost dissimilarity tensors $\mathbf{Z}^{(d)}$. *EHGM* uses the tensors differently depending on the degree d with respect to the branching step k.

3.5.3.1 Training

If the matching is being performed using the template method, the sample means $\bar{\mathbf{g}}_a^{(d)}$ are calculated during the model fitting step as they arise from training data. On the other hand, if matching is performed frame-to-frame, the previous frame hyperedge feature measurements are used in place of $\bar{\mathbf{g}}_a^{(d)}$, as the hypothesized feature measurements at evaluation are compared to the previous frame measurements.

At training, features are calculated across all N samples for the entire hypergraphical model. Mean vectors and inverse covariance matrices $\hat{\mathbf{\Sigma}}_a^{(d)}$ for all a and d are calculated at the training step for all dissimilarity tensors are calculated for each degree $d = 1, 2, \ldots, n_1$ hyperedge's feature set. The dissimilarity tensors $\mathbf{Z}^{(d)}$ are a function of a hypothesized matching and the sample estimates, and are the tensors thus only formed when evaluating a match.

3.5.3.2 Evaluation

At evaluation, the lower degree $d \leq 2k$ dissimilarity tensors are entirely computed prior to search. That is, each potential degree d matching in the new point set is compared to the reference hypergraph measurements. To detail further, each feature measurement per point-set and possible assignment is calculated and a the dissimilarity cost formed

and stored in the tensor $\mathbf{Z}^{(d)}$. Each potential assignment's hyperedge measurement $\mathbf{g}_a^{(d)}$ is calculated to construct the Mahalanobis distance cost. Lower degree ($d \leq 2k$) hyperedge dissimilarities are referenced in the precomputed $\mathbf{Z}^{(d)}$ tensors upon branching, whereas higher degree dissimilarities $d > 2k$ are calculated and stored on-the-fly on a per index basis.

The k-tuple of vertices assigned at branching and the accompanying second point-set serve as the input to calculate hyperedge features. The calculated degree d features of hyperedge a $\mathbf{g}_a^{(d)}$ describe that particular hyperedge dissimilarity: $(\mathbf{g}_a^{(d)} - \bar{\mathbf{g}}_a^{(d)})'(\hat{\mathbf{\Sigma}}_a^{(d)})^{-1}(\mathbf{g}_a^{(d)} - \bar{\mathbf{g}}_a^{(d)})$. The relevant indices of $\mathbf{Z}^{(d)}$ are calculated and stored when encountered during the search.

Each branching assigns k vertices. The hyperedge dissimilarities calculable upon the branching of degree $d \leq 2k$ are referenced while the higher degree $d > 2k$ terms are calculated as described.

3.5.4 Posture Identification in Embryonic *C. elegans*

Caenorhabditis elegans (*C. elegans*) is a small, free-living nematode found across the world. The worm is often studied as a model of nervous system development due to its relative simplicity [63, 89]. The adult worm features only 302 neurons, the morphology and synaptic patterning of which have been determined via electron microscopy [89]. The complete embryonic cell lineage has also been determined [77]; methods and technology have been developed to allow study of cell position and tissue development in the embryo [3, 11, 15, 52, 71, 83]. Systems-level studies of these processes may be able to discover

larger-scale principles underlying developmental events.

The embryo features a set of twenty *seam cells* and two associated neuroblasts. The seam cells and neuroblasts together describe anatomical structure in the coiled embryo, acting as a type of "skeleton" outlining its body. Identification of the seam cells and neuroblasts defines the embryo's posture. Fluorescent proteins are used to label cell nuclei, including the seam cell nuclei so that they may be visualized during imaging, e.g. with light sheet microscopy [90]. Volumetric images are captured at five minute intervals in order to capture subcellular resolution without damaging the worm's development [21]. Seam cell nuclei appear in the fluorescent images as homogeneous spheroids. Their positions relative to other nuclei and other salient cues present in the image volumes comprise the information that trained users employ to manually identify seam cells. Figure 3.6 shows the two rendered fluorescent images from Figure 3.1-A in Medical Image Processing, Analysis and Visualization (MIPAV), a 3D rendering tool [53]. The interface is used to annotate both seam cell nuclei and track remapped nuclei, as in Figure 3.3 [21].

We cast posture identification as hypergraph matching and use EHGM to solve the resulting optimization problem. The proposed models: *Sides*, *Pairs*, and *Posture* trade off modeling capacity for increased computation to identify optimal solutions. *Sides* expresses posture identification as graph matching; edge-wise (degree d=2) features take the form of standardized chord lengths between nuclei laterally and sequentially along each side. The first hypergraphical model, *Pairs*, employs a greater local context than *Sides* using degrees four and six hyperedges to describe relationships between seam cells. Hyperedges formed by two or three sequential pairs (d=4,6) better detail local regions throughout the embryo than is capable of a graphical model. Figure 3.7-A presents the hyperedge connectivity

among nodes in the *Pairs* model [81]. The *Posture* model extends the *Pairs* model by leveraging complete posture (d=n_1) features in effort to further discriminate between posture hypotheses that appear similar in sequential regions of the embryo. Geometric features help contextualize the coiled posture. Figure 3.8 illustrates three of the features used in the *Pairs* and *Posture* models. The angle Θ measures the angle between three successive pair midpoints. The angles Θ decrease throughout development as the worm elongates. Pair-to-pair twist angles φ and τ penalize posture hypotheses in which posterior to anterior transitions are jagged and unnatural in appearance. See A.1:*Posture Modeling* for further details and specification of model features.

The traditional point-set matching task requires a labelled point-set and a second unidentified point-set. Higher order features such as bend and twist angles may vary largely frame-to-frame depending on the posture at moment of imaging. However, elongation throughout late-stage development causes macroscopic trends in these geometric features. We estimate a template posture as a composite of feature measurements from a corpus of manually annotated postures. The templates are time dependent to reflect the elongation from the first point of imaging throughout development until hatching. See A.1:*Model Fitting* for details on template estimation.

Together, the fitted models are used with EHGM to identify posture in imaged *C. elegans* embryos. The branch size k=2 is set for all models, i.e. a lateral pair of seam cell identities are assigned at each branch starting with the tail pair cells *TL* and *TR*. The successive pair cells, *V6L* and *V6R*, are assigned given the established cells and hypergraphical relationships accessible with the hypothesized identities. Figure 3.9 depicts EHGM applied to the sample image depicted in Figure 3.1-A. The initial pair (*TL* and *TR*)

is selected, instantiating a search tree (Figure 3.9-A). Successive seam cell identities are partially assigned according to the given hypergraphical model in a pair-wise fashion. Each branch greedily queues hypothesized point-pair assignments conditioned on the previous branch assignments (black arrows within a branch). The next leading *V6* pair (Figure 3.9-E) is chosen upon exhaustion of the leading hypothesized *V6* pair (Figure 3.9-B). EHGM continues the recursion to implicitly identify a globally optimal posture under the given hypergraphical model; each possible initial pair will follow this illustrated process subject to pruning of the minimizing posture accessed via the hypothesized tail pair in Figure 3.9-A.

3.5.5 *EHGM* Posture Modeling

Embryonic *C. elegans* posture modeling used the aforementioned template hypergraph for quantifying hypothesized seam cell identities throughout the search process. The developing embryo elongates and as a result becomes more coiled due to the constraining eggshell. As such, the template hypergraphs are updated according to binned time intervals. Parameters are estimated from data according to the point in development between first image and hatch.

Each image is captured at time t with $n_1 = 20$ located nuclei centroids. The coordinates can be stored as $\mathbf{X} \in R^{n\times 3}$ which $\mathbf{X}_i = [x_i, y_i, z_i]$ representing the i^{th} centroid in R^3. The seam cells are ordered posterior to anterior: *TL*, *TR*, ..., *H0L*, *H0R*. Then let $1, 3, 5, \ldots 19$ be in the indices of the left side, and $2, 4, 6, \ldots 20$ be the indices of the right side. The seam cells are paired $(1, 2)$ for the tail pair, then $(2, 3), (4, 5), (6, 7), \ldots (19, 20)$ for the body pairs. Let $\mathbf{L} = (\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_{10})$ denote the left side nuclei locations, and similarly

R the right side nuclei locations.

The sampled worm embryos develop at similar rates. However, the occurrence of the first twitch, a point in development that triggers rapid physical changes, varies slightly embryo-to-embryo. As a result, we apply a time normalization in effort to compare feature measurements from images of the different specimens. Each embryo's time to first twitch is measured s_w, as well as hatch time h_w for sample embryo $w = 1, 2, \ldots 16$. The time points for each sample are indexed $k = 1, 2, \ldots n_w$. Each volume's imaging time t_{wk} is normalized to the $[0, 1]$ range via Eq 3.11. Each normalized time point z_{wk} is scaled such that $z_{wk} = 0$ represents first twitch, and $z_{wk} = 1$ hatching.

$$z_{wk} = \frac{t_{wk} - s_w}{h_w - s_w} \tag{3.11}$$

Features used in each of the three models: *Sides*, *Pairs*, and *Posture* are described with accompanying plots of their distributions throughout both development and within the embryo. All plots will feature the normalized time of observation z_{wk} on the horizontal axis. The vertical axis unit will vary between angular measurements, distances (in μm), and ratios of each which have no unit. Plots with multiple subplots in will measure features calculated at using segments of the embryo posterior to anterior. The left-most plot will depict calculations using the tail pair while the right-most will depict the feature ending with the *H0* pair at the anterior of the embryo.

3.5.5.1 *Sides*

The graphical model uses scaled distances between pairs of nuclei. The first feature to analyze is the distance between paired cell nuclei (A.1 Fig 3.10-A):

$$\mathbf{PD}_i = \|\mathbf{L}_i - \mathbf{R}_i\|_2 \tag{3.12}$$

The distance between the left and right seam cell nuclei within a pair can be interpreted as the width of the embryo measured at the sampled seam cell locations. The first (left-most) subplot of A.1 Fig 3.10-A illustrates distances in microns between nuclei centroids of the tail pair for each observation. The tail pair distance is used for the initial pair selection rule H_1 across all models. The second set of distances form along the left and right sides of the worm (A.1 Fig 3.10-B). The lengths of chords between successive nuclei on each side are calculated: $CL = \|\mathbf{L}_{i+1} - \mathbf{L}_i\|_2$ and $CR = \|\mathbf{R}_{i+1} - \mathbf{R}_i\|_2$. Similar to the pair distances, side length observations are highly variant.

The edge set: $\{(TL, TR), (TL, V6L), (TR, V6R), \ldots, (HOL, H0R)\}$ is expressed visually in Fig 3.11. The only features are the lengths between pairs of nuclear coordinates laterally and sequentially; i.e. there are 10 lateral edges and 18 sequential edges (9 on each side) yielding M_2=28 edges in total. The lateral pair edges: $\{(TL, TR), (V6L, V6R), (V5L, V5R), \ldots, (H0L, H$ are indexed $a = 1, 4, 7, \ldots, 25, 28$. Sequential edges on the left are indexed $a = 2, 5, 8, \ldots, 23, 26$ and sequential edges on the right are indexed $a = 3, 6, \ldots, 24, 27$. Each edge has $m_1 = m_2 = \ldots m_{28}$=1 feature.

3.5.5.2 *Pairs*

The *Pairs* model uses hyperedges connecting two or three pairs of nuclei (four or six nuclei). The first four features measure pair-to-pair variation, while the latter two features use triplets of pairs to measure angles formed by the midpoints of the three pairs. The first pair-to-pair feature extends upon the use of pair distances to better describe the coiled worm. The ratio of sequential pair distances models the variation in width throughout the assigned nuclei (Fig 3.12-A):

$$PDR_i = \frac{PD_i}{PD_{i+1}} \tag{3.13}$$

Each feature's estimated mean is slightly greater than 1, indicating that, on average, the worm is widening from tail to head. Another easily interpreted distance is the length of the chords connecting sequential pair midpoints. This is a more robust measure of worm length as side lengths vary more based upon the worm's folding (A.1 Fig 3.12-B):

$$MD_i = \|\mathbf{M}_{i+1} - \mathbf{M}_i\|_2 \tag{3.14}$$

The length of the chords connecting sequential pair midpoints is a more robust measure of worm length as side lengths vary more based upon the worm's folding. The cosine similarity is used to assess the degree to which sequential sides are pointing in the same direction (A.1 Fig 3.12-C):

$$\phi_i = \frac{(\mathbf{R}_{i+1} - \mathbf{R}_i) \cdot (\mathbf{L}_{i+1} - \mathbf{L}_i)}{\|\mathbf{R}_{i+1} - \mathbf{R}_i\|_2 \|\mathbf{L}_{i+1} - \mathbf{L}_i\|_2} \in [-1, 1] \tag{3.15}$$

The final two pair-to-pair *Pairs* features attempt to model two different types of *twist* in the posture. The lateral and axial twists measures angles of rotation from lateral and posterior views, respectively (A.1 Fig 3.12-D).

$$\mathbf{b}_1 = \frac{\mathbf{L}_{i+1} - \mathbf{L}_i}{\|\mathbf{L}_{i+1} - \mathbf{L}_i\|_2} \tag{3.16}$$

$$\mathbf{b}_2 = \frac{\mathbf{L}_i - \mathbf{R}_i}{\|\mathbf{L}_i - \mathbf{R}_i\|_2} \tag{3.17}$$

$$\mathbf{b}_3 = \frac{\mathbf{R}_i - \mathbf{R}_{i+1}}{\|\mathbf{R}_i - \mathbf{R}_{i+1}\|_2} \tag{3.18}$$

$$\mathbf{b}_4 = \frac{\mathbf{R}_{i+1} - \mathbf{L}_{i+1}}{\|\mathbf{R}_{i+1} - \mathbf{L}_{i+1}\|_2} \tag{3.19}$$

$$\mathbf{n}_1 = \mathbf{b}_1 \times \mathbf{b}_2 \tag{3.20}$$

$$\mathbf{n}_2 = \mathbf{b}_2 \times \mathbf{b}_3 \tag{3.21}$$

$$\mathbf{n}_3 = \mathbf{b}_3 \times \mathbf{b}_4 \tag{3.22}$$

$$\mathbf{c}_1 = < \mathbf{n}_1 \times \mathbf{n}_2, \mathbf{b}_2 > \tag{3.23}$$

$$\mathbf{c}_2 = < \mathbf{n}_1, \mathbf{n}_2 > \tag{3.24}$$

$$\psi_i = \frac{1}{\pi} atan2(< \mathbf{n}_1 \times \mathbf{n}_2, \mathbf{b}_2 >, < \mathbf{n}_1, \mathbf{n}_2 >)) \tag{3.25}$$

$$\tag{3.26}$$

Axial twists present between a sequence of two pairs calculates the angle obtained by projecting the chord linking pairs onto each other (A.1 Fig 3.12-E):

$$\mathbf{b}_1 = \frac{\mathbf{L}_i - \mathbf{L}_{i+1}}{\|\mathbf{L}_i - \mathbf{L}_{i+1}\|_2} \tag{3.27}$$

$$\mathbf{b}_2 = \frac{\mathbf{R}_i - \mathbf{L}_i}{\|\mathbf{R}_i - \mathbf{L}_i\|_2} \tag{3.28}$$

$$\mathbf{b}_3 = \frac{\mathbf{R}_{i+1} - \mathbf{R}_i}{\|\mathbf{R}_{i+1} - \mathbf{R}_i\|_2} \tag{3.29}$$

$$\mathbf{b}_4 = \frac{\mathbf{L}_{i+1} - \mathbf{R}_{i+1}}{\|\mathbf{L}_{i+1} - \mathbf{R}_{i+1}\|_2} \tag{3.30}$$

$$\mathbf{n}_1 = \mathbf{b}_1 \times \mathbf{b}_2 \tag{3.31}$$

$$\mathbf{n}_2 = \mathbf{b}_2 \times \mathbf{b}_3 \tag{3.32}$$

$$\mathbf{n}_3 = \mathbf{b}_3 \times \mathbf{b}_4 \tag{3.33}$$

$$\mathbf{c}_1 = < \mathbf{n}_2 \times \mathbf{n}_3, \mathbf{b}_3 > \tag{3.34}$$

$$\mathbf{c}_2 = < \mathbf{n}_2, \mathbf{n}_3 > \tag{3.35}$$

$$\tau_i = \frac{1}{\pi} atan2(< \mathbf{n}_2 \times \mathbf{n}_3, \mathbf{b}_3 >, < \mathbf{n}_2, \mathbf{n}_3 >)) \tag{3.36}$$

$$\tag{3.37}$$

Angles along sides of the worm formed by triples of sequential nuclei approximate bend in the worm along each side. These bend angles are highly variant, especially frame-to-frame, in the same manner as side lengths in *Sides* (A.1 Fig 3.10-B). Angles formed by pair midpoints exacerbate the computational burden as six nuclei are required, compare to three in a typical angle calculation, but the midpoint based angles are less variant than angles of each side (A.1 Fig 3.12-F):

$$\Theta_i = \frac{180}{\pi} \arccos \frac{< \mathbf{M}_{i+1} - \mathbf{M}_i, \mathbf{M}_{i+2} - \mathbf{M}_{i+1} >}{\|\mathbf{M}_{i+1} - \mathbf{M}_i\|_2 \|\mathbf{M}_{i+2} - \mathbf{M}_{i+1}\|_2} \tag{3.38}$$

Each angle $\Theta \in [0, 180]$ where 0 would denote the worm perfectly folded upon itself, and 180 would define a flat worm. A second set of angles aims to approximate the posterior to anterior bend in the worm. The angles ζ_i are defines as the angles formed by fitted planes intersecting between pair midpoints (A.1 Fig 3.12-G):

$$\zeta_i = \frac{180}{\pi} \frac{< (\mathbf{R}_{i+1} - \mathbf{L}_{i+1}) \times (\mathbf{M}_{i+1} - \mathbf{M}_i), ((\mathbf{R}_{i+1} - \mathbf{L}_{i+1}) \times (\mathbf{M}_{i+1} - \mathbf{M}_{i+1})) >}{\|(\mathbf{R}_{i+1} - \mathbf{L}_{i+1}) \times (\mathbf{M}_{i+1} - \mathbf{M}_i)\|_2 \|(\mathbf{R}_{i+1} - \mathbf{L}_{i+1}) \times (\mathbf{M}_{i+1} - \mathbf{M}_{i+1})\|_2} \tag{3.39}$$

Fig 3.7-A depicts the *Pairs* hyperedge set:

$$\{(TL, TR), (TL, TR, V6L, V6R), \ldots, (H1L, H1R, H0L, H0R)\}.$$

An initial degree d=2 edge ($\{(TL, TR)\}$, M_1=1) accompanies degrees d=4 and d=6 hyperedges built sequentially throughout the embryo. There are M_4=9 degree 4 hyperedges:

$$\{(TL, TR, V6L, V6R), (V6L, V6R, V5L, V5R), \ldots, (H1L, H1R, H0L, H0R)\}$$

and M_6=8 degree 6 hyperedges:

$$\{(TL, TR, V6L, V6R, V5L, V5R), \ldots, (H2L, H2R, H1L, H1R, H0L, H0R)\}.$$

The degree two feature $g_1^{(2)} = PD$ (m_1=1) is used for the initial branch selection rule. Then, $g_a^{(4)} = [PDR, MD, \phi, \psi, \tau]$ ($m_1 = m_2 = \cdots = m_9$=5 for d=4), $a = 1, 2, \ldots, 9$ are calculated for degree d=4 hyperedge traversing the embryo, posterior to anterior. Finally, the $g_a^{(6)} = [\theta, \zeta]$ ($m_1 = m_2 = \cdots = m_8$=2 for d=6), $a = 1, 2, \ldots, 8$ are calculated for degree d=6 hyperedges. Each degree 4 hyperedge uses the same features while each degree 6 hyperedge uses the same features.

3.5.5.3 *Posture*

The *Posture* model is comprised of all *Pairs* features as well as the features defined by the summations of each local feature measurement throughout the hypothesized posture. Full posture features give insight into the changes in the embryo's shape throughout late-stage embryogenesis (A.1 Fig 3.13). Worm length follows an approximately logarithmic pattern.Total curvature follows a negative exponential pattern. Earlier on the worm is fatter and cannot bend as much. The worm elongates during development, allowing for sharper bends.

Fig 3.14 depicts the *Pairs* hyperedge set with the full degree n_1-20 hyperedge:

$$\{(TL, TR, V6L, V6R, V5L, V5R, \ldots, H2L, H2R, H1L, H1R, H0L, H0R)\}$$

. The *Pairs* edge and hyperedge set is duplicated with an additional degree n_1 hyperedge: $(TL, TR, \ldots, H0L, H0R)$. The degree n_1=20 hyperedge (M_{20}=1) features are summations of the *Pairs* features. The degree n_1=20 hyperedge (M_{20}=1) has m_1=7 features, the summations of the degrees 4 and 6 features of *Pairs*.

3.6 Acknowledgments

This work utilized the computational resources of the NIH HPC Biowulf cluster. (http://hpc.nih.gov). Dr. Evan Ardiel was instrumental in developing descriptive features for identifying worm posture. Post-Baccalaureate research fellows Brandon Harvey and Nensi Karaj were supportive in providing data and discussions concerning the modeling. Dr. Zhen Zhang and Dr. Arye Nehorai provided assistance in using *KerGM* [96]. Dr. Vincent Lyzinski also provided insight on the methods. We also thank Dr. Hank Eden and Dr. Matthew Guay for their careful readings and suggestions. The code and data are available at `https://github.com/lauziere/EHGM`.

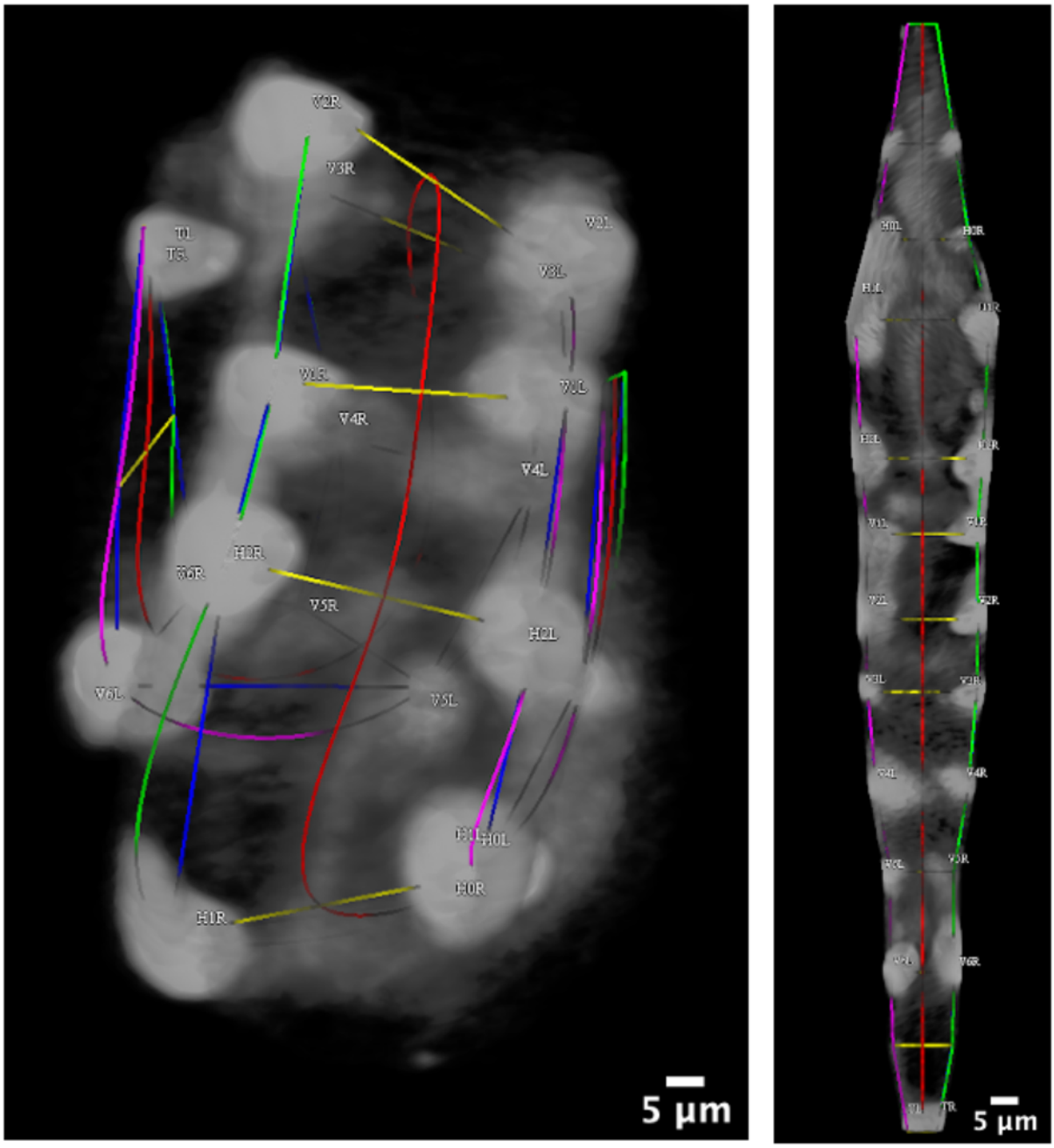

Figure 3.6: **Rendered image volumes in the MIPAV GUI.** The imaged twisted embryo (left) and imaged straightened embryo (right) rendered in Medical Image Processing, Analysis and Visualization (MIPAV) [53]. The fluorescent images are those depicted in Figure 3.1-A. Trained users navigate the MIPAV GUI to identify seam cells based upon relative positioning and other salient features such as specks of fluorescence on the skin. Correct identification of all imaged nuclei reveals the coiled embryonic posture. Green (left), red (center), and purple (right) splines yield an approximation of the coiled embryo's posture. Yellow lines connect seam cell nuclei laterally. The splines are used with the image volume to sweep planes orthogonal to the center spline, yielding the straightened embryo image.

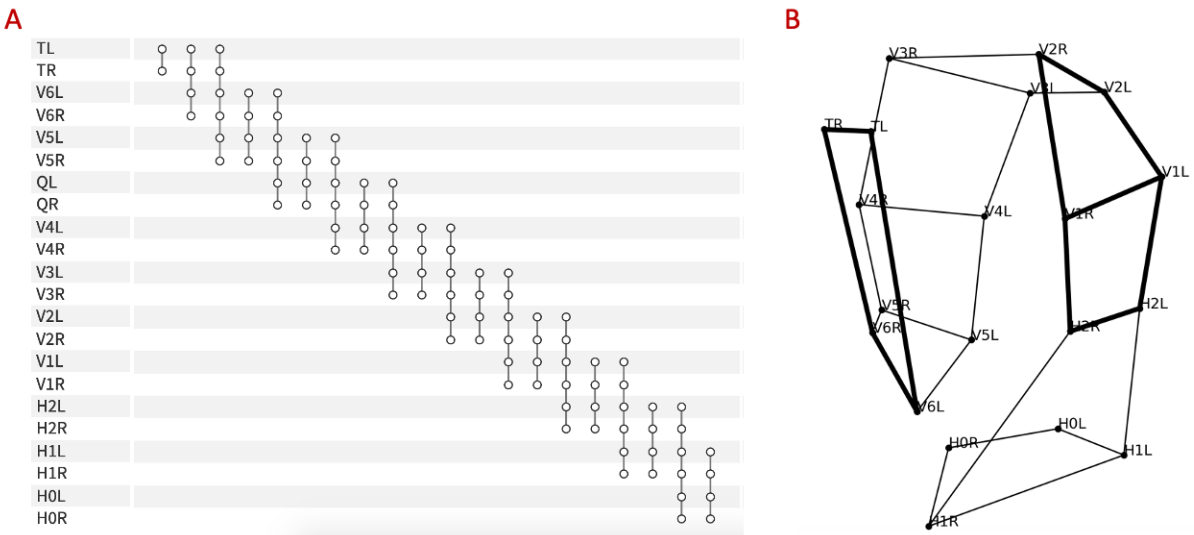

Figure 3.7: **The *Pairs* hypergraphical model uses expansive local contexts about each portion of the embryo.** A: The *Pairs* hyperedges connect local seam cell nuclei in sets of four and six. B: Degree four hyperedges connect sequential pairs of seam cells while degree six hyperedges connect sequential triplets of pairs. The posterior-most degree four hyperedge and a central degree six hyperedge are **bolded**.

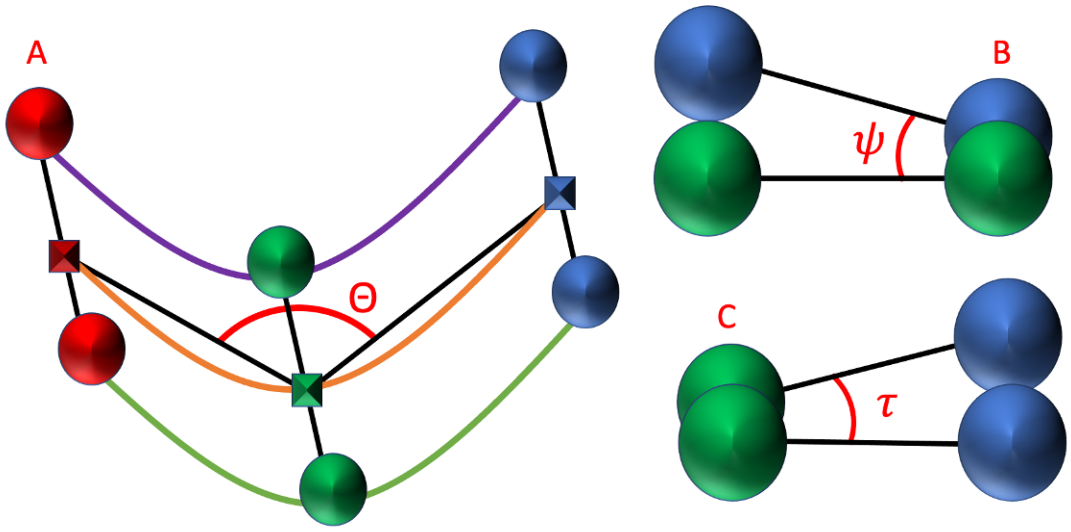

Figure 3.8: **Hypergraphical geometric features contextualize seam cell assignments.** Anatomically inspired geometric features describe bend and twist of a posture assignment. A: Three pairs of sequential nuclei: red, green, blue. Rectangles represent pair midpoints. The angle Θ in red is used as a degree six feature given six point to nuclei assignments. B, C: Degree four hypergraphical features measuring twist angles φ and τ. These angles measure posterior to anterior twist pair-to-pair and left-right twist, respectively.

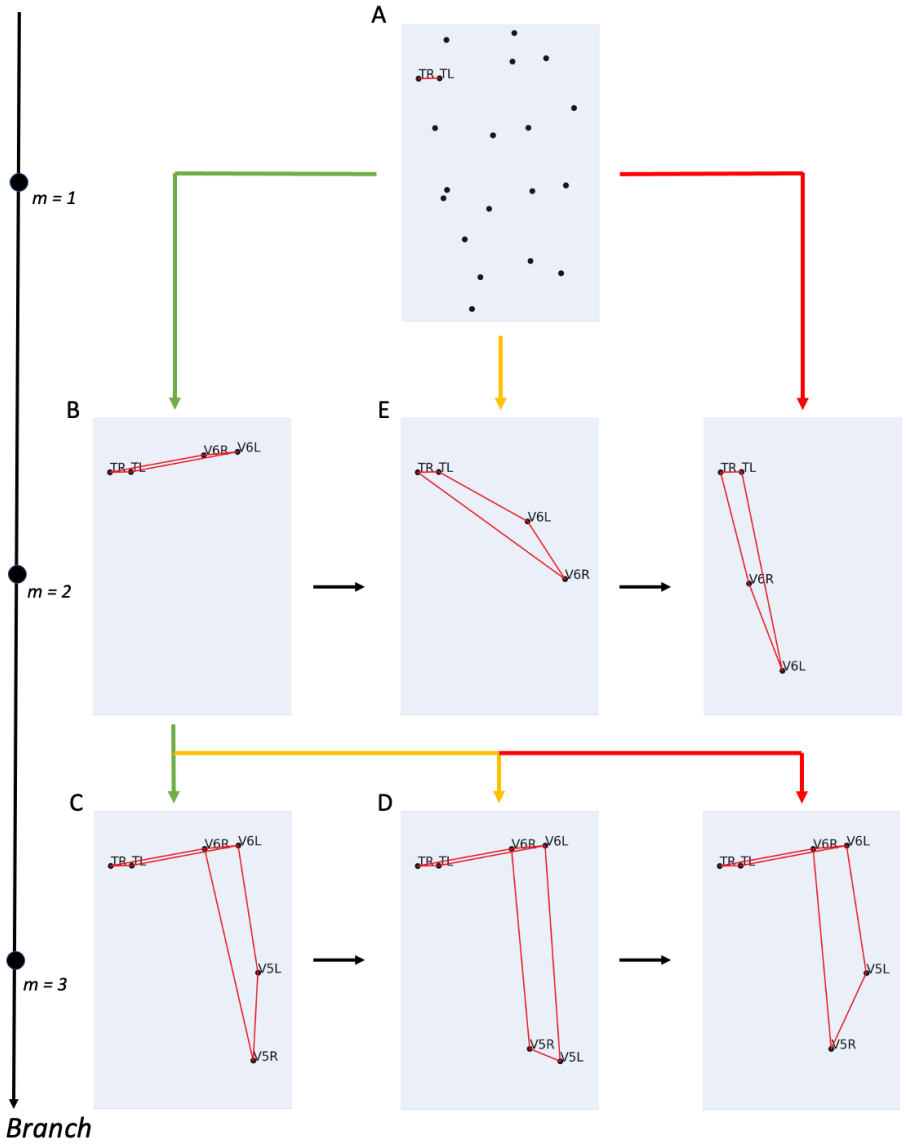

Figure 3.9: **EHGM applied to the sample image depicted in Figure 3.1-A.** A: Two points are selected at the initial branch for *TL* and *TR*, respectively. Candidates for the successive pair, *V6L* and *V6R*, are queued based on hypergraphical relationships between the established cell identities *TL* and *TR* and each hypothesized *V6* pair (lower costs are green to higher costs in red). B: The leading hypothesis at branch m=2 given the initial branch pair is chosen. The recursion continues to queue *V5* pair choices at branch m=3. Black arrows within branch m specify the ordering of the branch given established cell assignments. Each branch creates a new subproblem of completing the posture given partially assigned identities. C: The tree continuing from the *V5* pair hypothesis is fully explored according to the established recursion. D: The next leading *V5* hypothesis is initiated upon exhaustion of the subtree formed at panel C. E: Implicit enumeration of the subtree formed at panel B causes the search to progress to the second leading *V6* hypothesis.

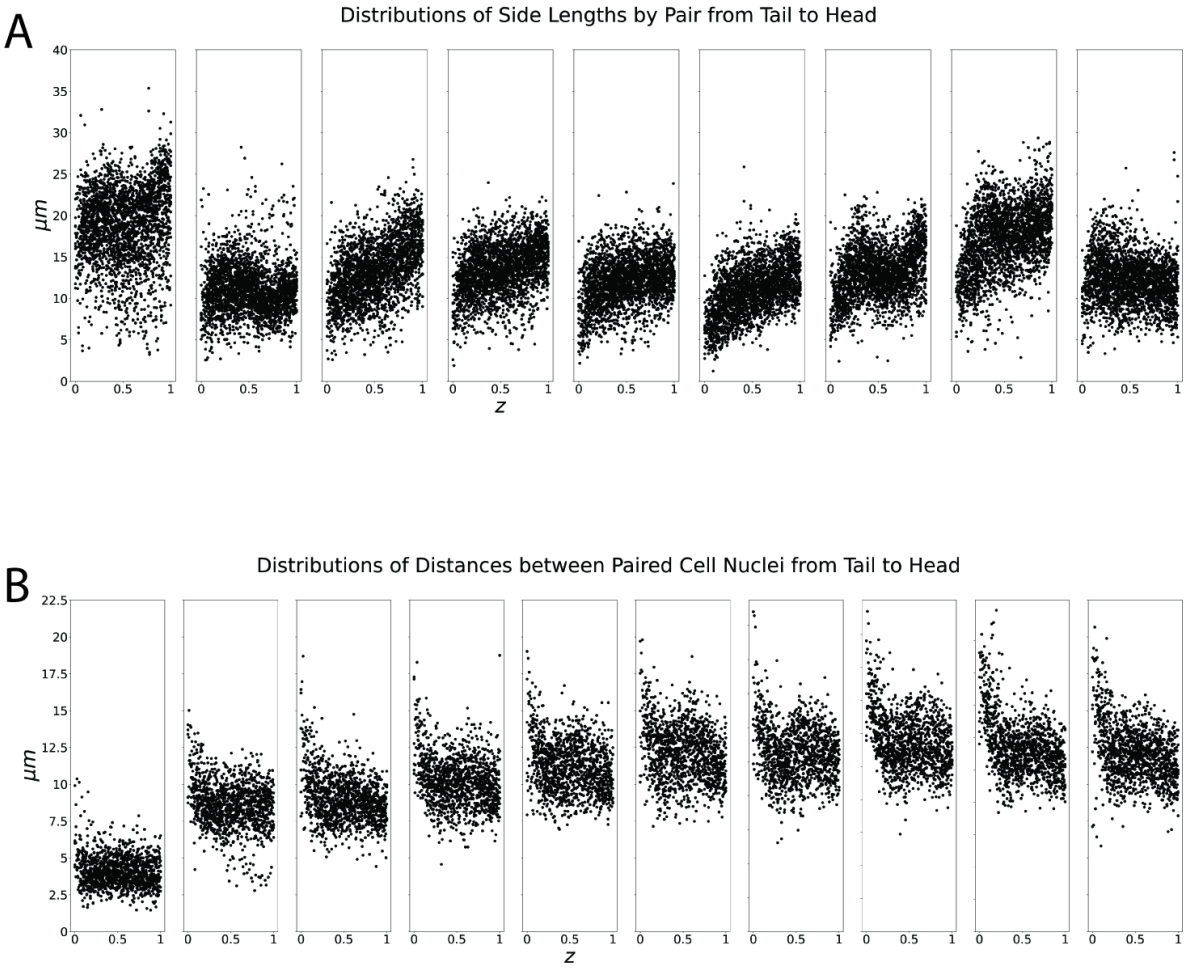

Figure 3.10: ***Sides* model features.** A) Distances between nuclei of lateral pairs. Notably, the tail pair distance (left-most panel) is constant throughout imaging. The tail pair distance informs the initial pair selection rule H_1. B) Chord lengths along left and right sides of the posture. Both quadratic features show high variance.

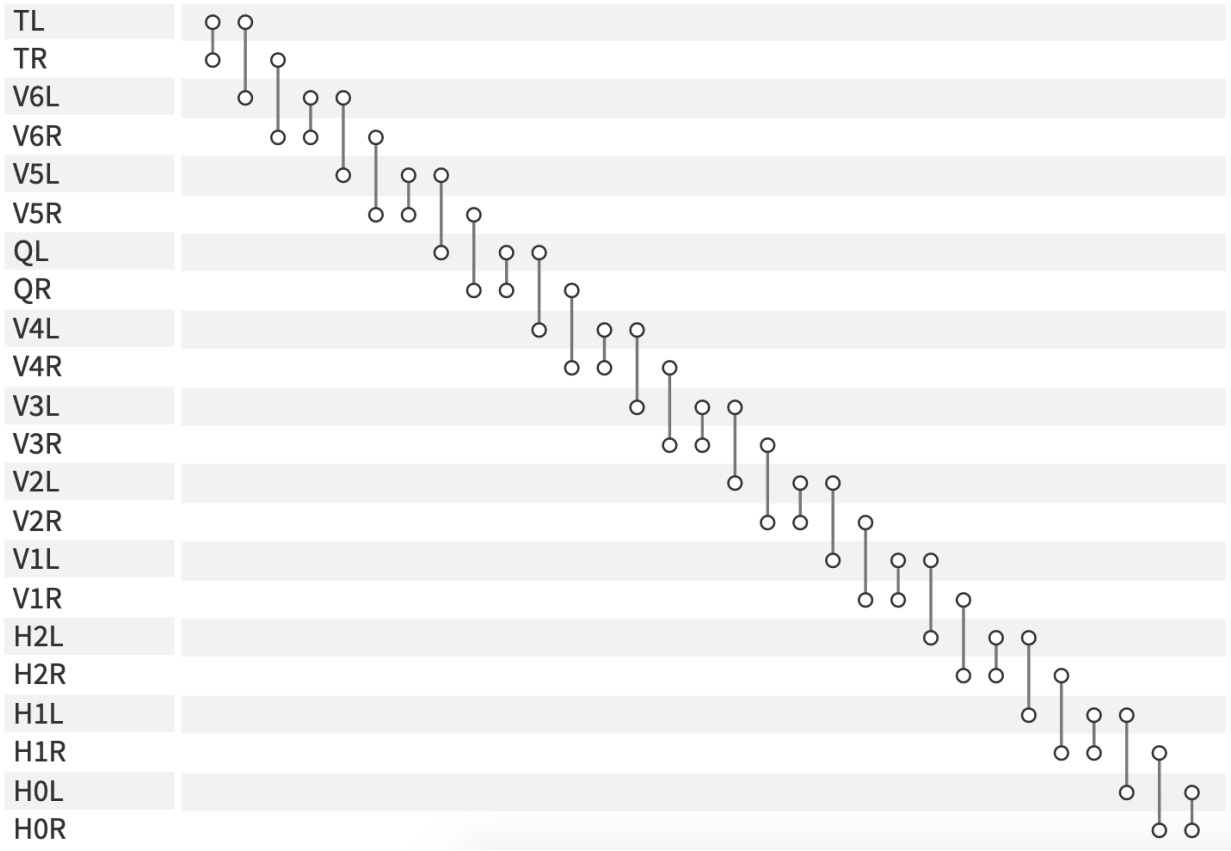

Figure 3.11: ***Sides* model edge set.** Edge connect pairs of cells laterally and sequentially along each side.

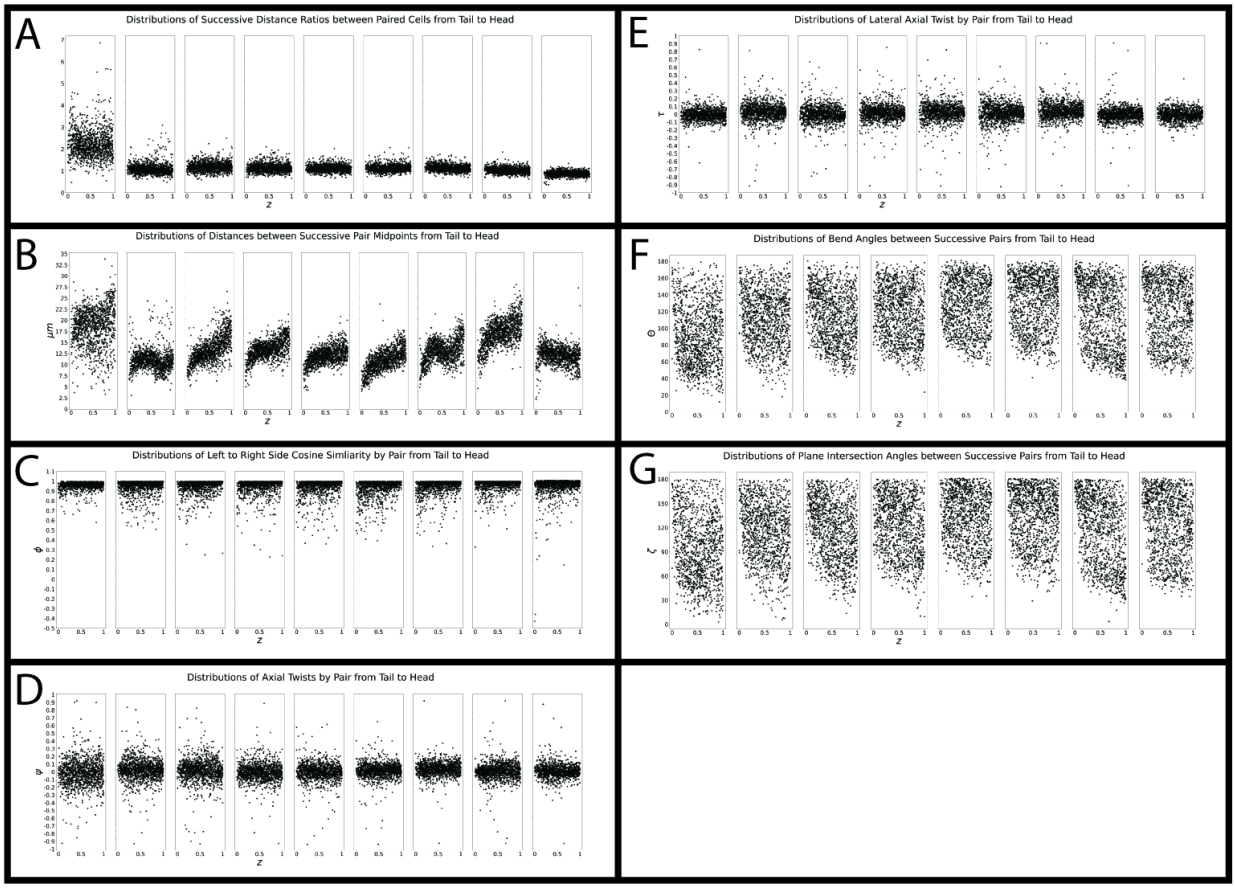

Figure 3.12: ***Pairs* model features.** A) Ratios of pair distances (Eq 3.13). B) Distance between successive pair midpoints (Eq 3.14). C) Cosine similarities between successive left and right sides (Eq 3.15). D) Lateral axial twist angles (Eq 3.16). E) Axial twist angles (Eq 3.27). F) Midpoint bend angles (Eq 3.38). G) Planar intersection angles (Eq 3.39).

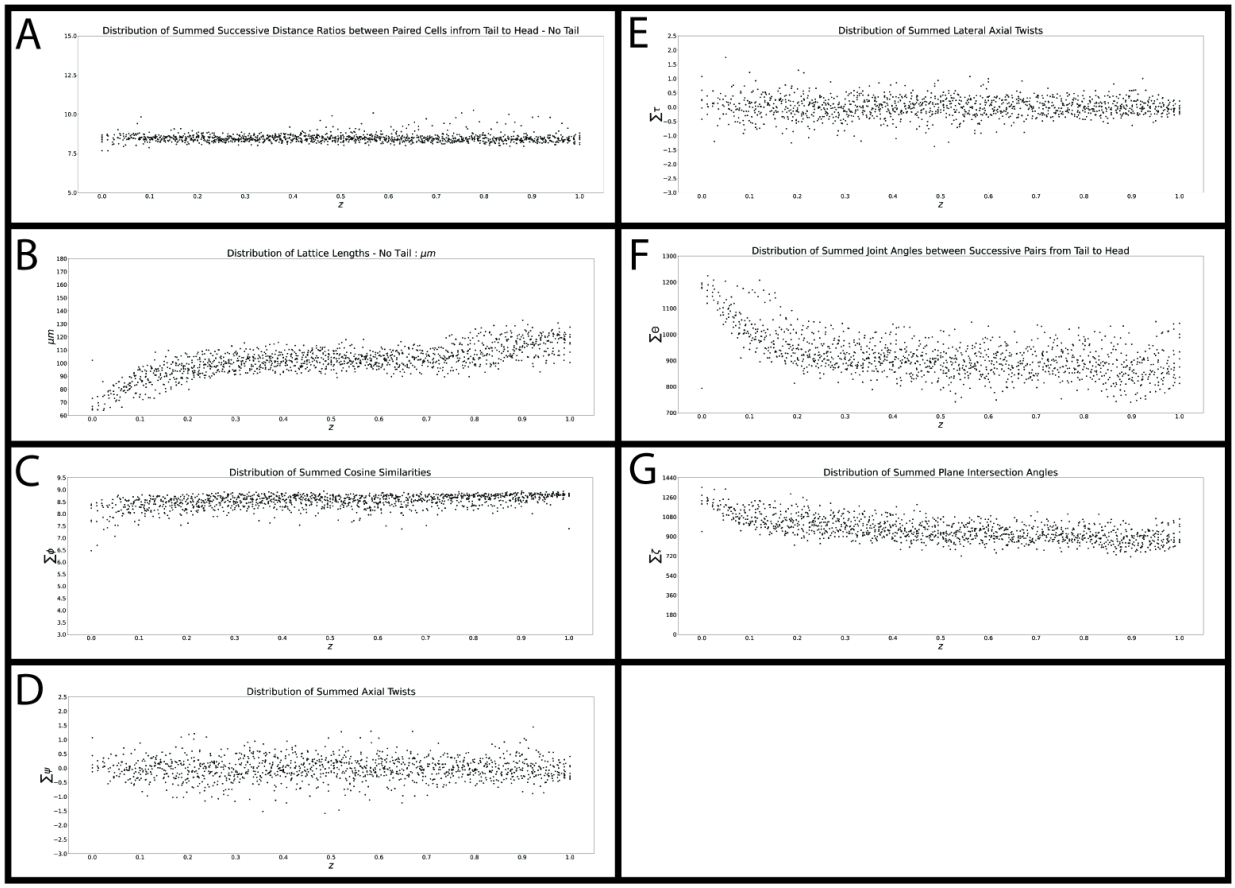

Figure 3.13: ***Posture* model features include all *Pairs* features and posture-wide versions of *Pairs* features.** A) Summed ratios of pair distances (Eq 3.13). B) Summed distances between successive pair midpoints (Eq 3.14). C) Summed cosine similarities between successive left and right sides (Eq 3.15). D) Summed lateral axial twist angles (Eq 3.16). E) Summed axial twist angles (Eq 3.27). F) Summed midpoint bend angles (Eq 3.38). G) Summed planar intersection angles (Eq 3.39).

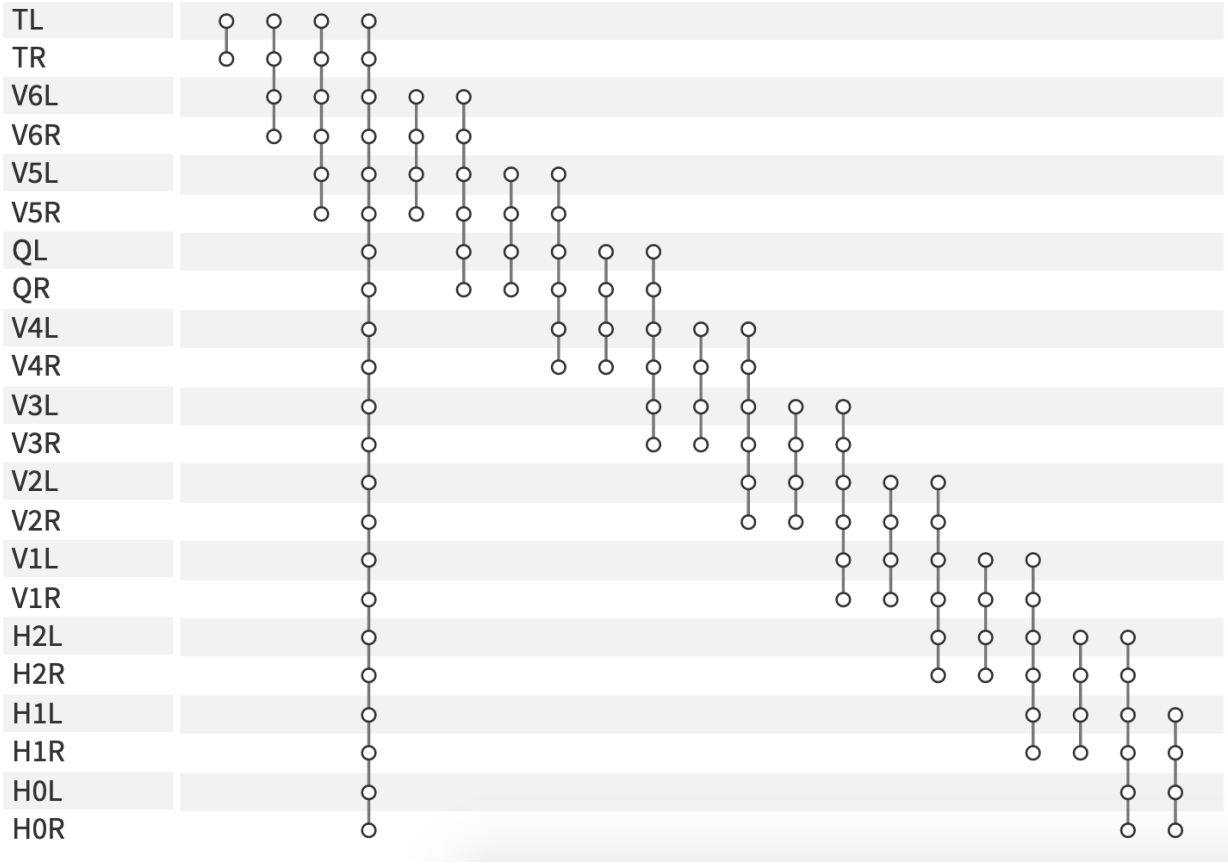

Figure 3.14: ***Posture* model hyperedges.** *Posture* hyperedges include all *Pairs* hyperedges and posture-wide hyperedge.

Chapter 2: A Semi-automatic Cell Tracking Process Towards Completing the 4D Atlas of *C. elegans* Development

2.1 Introduction

The 4D atlas has been completed up to twitching as the embryo is relatively still prior to the development of muscle cells [77]. Posture identification serves as an intermediate step towards cell tracking in late-stage development due to the muscular twitching [21]. The post-twitch embryo remains a challenge to established cell tracking methodology [71]. The progression of individual cells to tissues to phenotypic functions across systems can be observed with completion of the full developmental atlas [21].

Current approaches to cell tracking in the post-twitch embryo require the identification of seam cells to approximate the coiled posture (Chs. 1.2, 3.5.4). In brief, the seam cells and neuroblasts together describe anatomical structure in the coiled embryo, acting as a type of "skeleton" outlining its body. *Untwisting* the worm describes the computational process of using the identified seam cell nuclei coordinates to remap coordinates of other imaged cells to a straightened coordinate system [21]. Subsets of cells can be imaged, detected, remapped, then tracked. Figure 3.2-C illustrates the untwisting process on a magnified portion of the right coordinate set in Figure 3.2-A. The remapping uses the left, right, and

midpoint splines to establish a change of basis to a straightened coordinate space defined by the tangent (black), normal (blue), and binormal (cyan) vectors. Nuclei encountered are remapped according to the position along the spline and location within the closest inscribed ellipse. Black lines connect cell nuclei in the left frame (red) to their locations in the middle frame (blue); muscle cell names arise from the four bands (A-D), anterior to posterior (Figure 3.2-D).

The presented strategy is still dependent on trained users to manually identify and track nuclei in an image rendering GUI [21, 53]. However, the resulting "untwisted" images yield a feasible multiple object tracking (MOT) scenario. The movement of cell nucleus i from frame *t-1* to t can be approximately decomposed:

$$\|\mathbf{x}_i^{(t)} - \mathbf{x}_i^{(t-1)}\|_2 = \Delta D_i^{(t)} \approx \Delta^I D_i^{(t)} + \Delta^O D_i^{(t)} \tag{2.1}$$

where $\Delta^I D_i^{(t)}$ measures movement of nucleus i *inside* the embryo and $\Delta^O D_i^{(t)}$ measures nuclear movement attributed to the embryo repositioning between images. The untwisting process mitigates $\Delta^O D_i^{(t)}$, movement attributed to the embryo moving. The remaining term $\Delta^I D_i^{(t)}$ can be modeled with MOT methods. However, the untwisting process is imperfect; the seam cells represent an approximation of the coiled embryo. Users in the rendering GUI add focal points to the spline fitting to more accurately pinpoint nuclei locations in the untwisted image. The manual effort is mitigated with a proposed MOT method which is integrated to the GUI.

2.2 Methods

Frame-to-frame displacement of cell i at time t: $\Delta D_i^{(t)}$ is unpredictable due to the embryo repositioning between five minute images. The seam cells are used to estimate complete cubic splines through the left and right sides of the embryo [21]. The remapped coordinates (Figure 3.2) mitigate the displacement attributed to embryo movement $\Delta^O D_i^{(t)}$, leaving only the *inter*-worm cell movement $\Delta^I D_i^{(t)}$. This movement can be assumed to be Brownian between frames, allowing for the application of MOT strategies to track "straightened" nuclei. Sample data of the 85 muscle nuclei (Figure 3.2-B) are used to evaluate the efficacy of automated tracking approaches.

2.2.1 Nuclei Detection

Detect-and-track MOT paradigms (Chapter 2.2) rely to finding objects in images then associating unique objects between frames. Nuclei are detected in the sample images and compared to accompanying annotated coordinate sets. Convolutional neural networks (CNNs) achieve state of the art performance in image segmentation tasks. Two approaches are evaluated in this exploration: a 3D U-Net [69, 98] and *Stardist 3D* [85, 86]. The 3D U-Net achieves the best results in image segmentation. Two 3D U-Net models are trained either from a random initialization or from similar data and of differing sizes to achieve voxel-level segmentations. The architecture itself remains constant, but the number of filters in each layer varies according to each model. The L model has twice as many filters in each layer as M, which has twice as many filters as S. 3D U-Net models are trained to maximize the dice coefficient via the Adam optimization scheme [37]. *Stardist 3D* combines

elements of a FCN such as the 3D U-Net but with a focus on identifying disjoint objects. Two configurations of *Stardist 3D* are evaluated; the first uses a pretrained model trained on seam cell nuclei while the second is trained from a random initialization.

2.2.2 Nuclei Tracking

The simplest paradigm for MOT is to solve sequential frame-to-frame GNN LPs (Eq 2.9). The random motion *within* the embryo is amenable to the L2 norm cost; however, the lack of trajectory to cellular motion invalidates a dynamical model such as the Kalman Filter. However, the coordinate remapping is often imperfect and may shift nuclei from their true locations due to the spline curves not perfectly adhering to the embryo body wall. A graph can specify interactivity among adjacent nuclei, allowing for a stronger representation of nuclear movement than is possible with the GNN. Graph matching (Chapter 2.1.2) may provide stronger results, but the computation required due to the high number of muscle nuclei forces the use of heuristic algorithms. Kernelized Graph Matching (KerGM) [96] is the most recent development in heuristic graph matching and is applied to track straightened muscle nuclei. While GNN based solutions have been adapted for object disappearance, reappearance, merging, and splitting [22, 34], KerGM and such graphical methods have not been adapted. Indeed, the methods can perform one-to-many or many-to-one assignment, but the methods cannot handle the common scenario in which a cell disappears in one image, and a different cell appears in the subsequent image.

However if an LAP yields strong solutions, then evaluating multiple solutions via a more complex cost could improve the tracking quality. Murty's algorithm allows for

returning the K best solutions to an LAP with complexity $\mathcal{O}(Kn^3)$ [54, 55]. The K hypothesized solutions to the LAP can then be evaluated on a quadratic cost. Evaluating a higher order cost is computationally inexpensive compared to searching the entire space for the assignment that minimizes the quadratic cost. If the LAP returns high quality solutions, then sampling and evaluating at the QAP cost could improve performance by further applying a more complex model to discriminate between competing hypotheses.

Both *KerGM* and the evaluation of a graph cost in Murty's algorithm require a specified graph for each detection set. The choice in graph greatly influences the results. A QAP with no edge-wise connections reduces to an LAP. On the other hand, an overly dense edge set may not generalize well to unseen data. Thus several types of graphs are evaluated. The first four graphs connect nuclei in the straightened space that are withing $5\mu m$, $7.5\mu m$, $10\mu m$, and $12.5\mu m$ respectively. Delaunay Triangulation is used as another method of generating a graphical representation of the nuclei in the straightened space at each image [5].

2.2.3 MIPAV Interative Tracking Plugin

Sequential detection and matching allows for accurate frame-to-frame tracking of the straightened coordinates. However, the simple matching paradigm is susceptible to large errors from false positives and false negatives. Near perfect detections are necessary for an effective frame-to-frame matching paradigm. The Untwisting plugin [21] for MIPAV [53] was augmented to perform semi-automatic frame-to-frame tracking. The 3D rendering allows for interactive point identification and the import and export of such data. An

initial detection set is superimposed on the volume in a 3D viewer. The user is able to perfect the detection set by removing false detections, adding missed nuclei, and splitting erroneously clumped nuclei. A first pass of the matching model links detections in the next frame to named nuclei in the prior. The user can then make corrections if necessary, and then resolve the linear program problem given user verified matched nuclei being passed as encoded constraints. This process can be repeated to completion in real time due to the polynomial complexity of solving the GNN LP [35].

Figure 2.1 illustrates the semi-automated process. The seed volume detections are assessed and nuclei are identified (step 1). Then, the sequential volume detections are similarly assessed to yield an accurate detection set (step 2). Steps 3A and 3B form a recursion whereby the user generates predicted identities of the second volume nuclei (step 3A) and then edits the predicted nuclear identities. Another round of matching with added constraints will regenerate predictions (step 3B). The process is expedited with accurate detections.

2.3 Results

A set of $n = 233$ images with densely labelled cell nuclei are manually annotated to train a model to better segment smaller, closer cell nuclei. The n pairs of fluorescently imaged embryos and binary annotation masks are used to fit segmentation models described in Chapter 2.2.1.

Images from three embryos bred to illuminate the muscle cell nuclei were annotated for evaluating the tracking methodology. The strain *KP9305* targets the four bands of

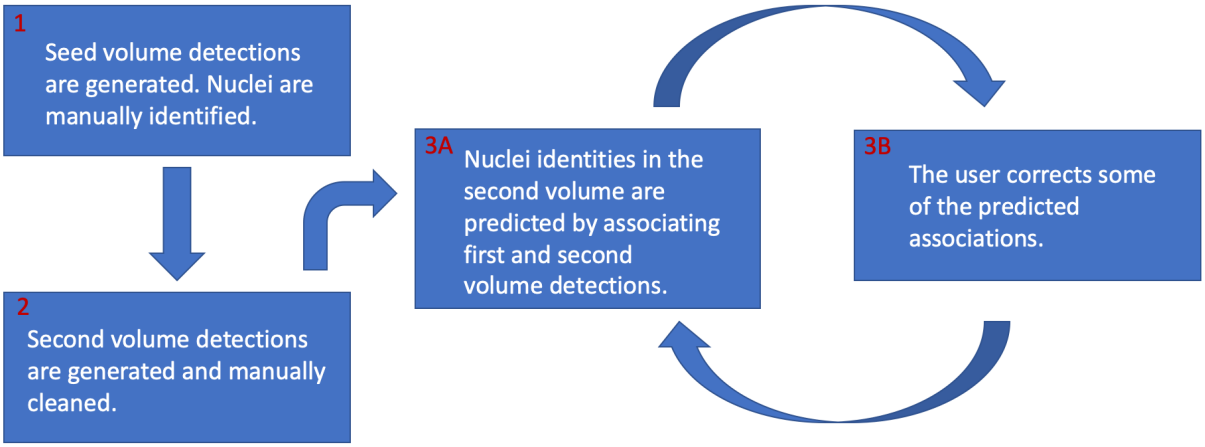

Figure 2.1: **A recursive cell tracking enables real-time cell identification in the straightened embryo.** The semi-automatic densely labelled nuclei tracking paradigm. Sets of detections in the second fluorophore are remapped via the seam cell lattice. The initial seed detections are identified (step 1). Then, the sequential frame detections are simply edited to account for all nuclei in the image (step 2). The recursive tracking procedure iteratively corrects the nuclear alignments in a manner designed to minimize the manual burden.

muscle cell nuclei within the worm embryo, each band composing approximately 21 muscle nuclei. The 85 nuclei are to the best ability of the researchers identified in each image volume throughout development for each of the three sampled worms. Nuclei centroids with associated identity are provided for all expertly detected nuclei in each image across the three worms. Figure 2.2 depicts XY maximum projections from two sequential image volumes. Cell nuclei homogeneity and density contribute to the tracking challenge.

2.3.1 Detection

Voxel-wise evaluation does not adequately measure the performance of nuclei detection models. Each model and postprocessing combination yields a set of detections to be compared to annotation volume nuclei sets. 3D U-Net models [69, 98] are compared to

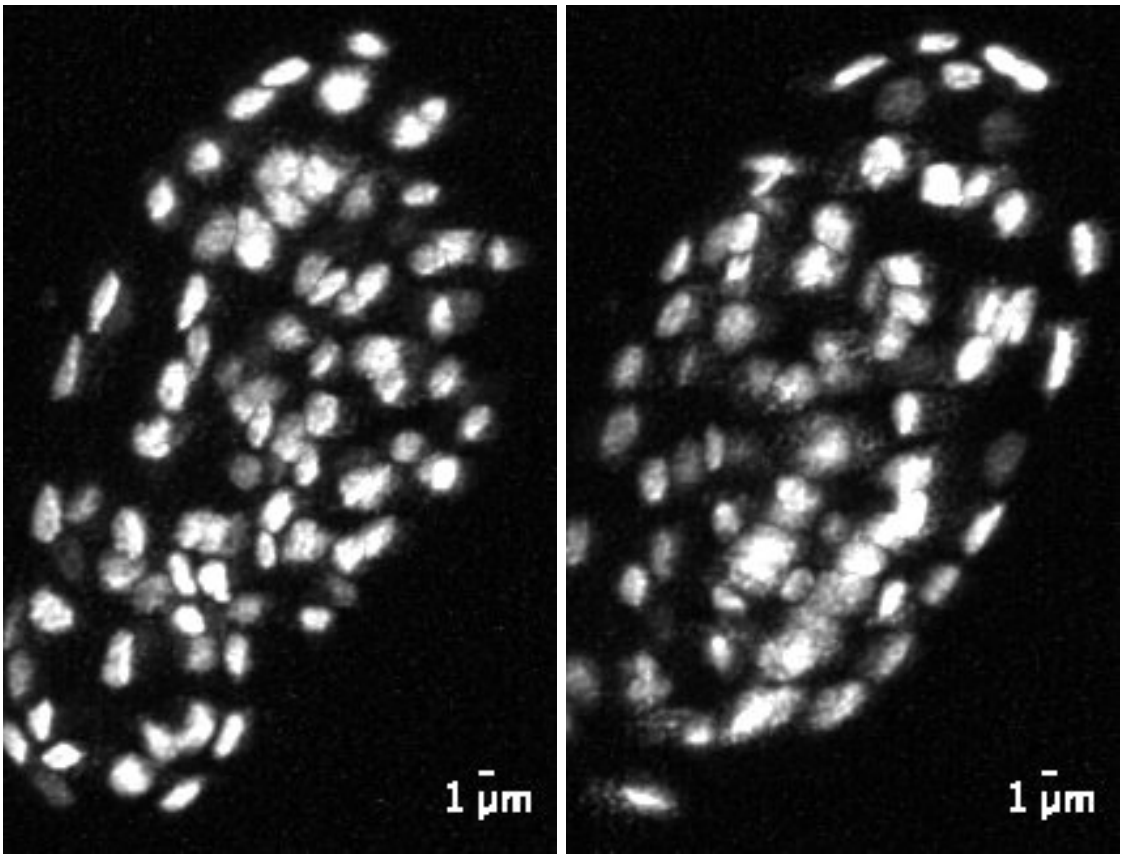

Figure 2.2: **Volumetric fluorescent images expressing fluorescence in 81 muscle cell nuclei.**XY maximum intensity projections of two sequential image volumes. The images come from the first worm embryo, approximately four hours prior to hatching. The nuclei are each inside muscle cells within the worm embryo.

Stardist 3D [85, 86]. Table 2.1 highlights nuclei centroid matching results from processed image volumes in the held-out test set. *Stardist 3D* is the most precise, but tends to miss more dim nuclei than the 3D U-Net models. The nuclei that can appear extremely close vary in size and intensity. Building an effective model for nuclear identification remains a barrier to automatic cell tracking.

	Precision	Recall	F1
Stardist 3D	0.81	0.63	0.68
3D U-Net - S	0.64	0.67	0.65
3D U-Net - M	0.69	0.77	0.72
3D U-Net - L	0.67	0.75	0.70

Table 2.1: **Dense nuclei detection is challenging despite the application of state-of-the-art CNNs.** The nuclei are smaller and tend to be more clustered upon each other. *Stardist 3D* excels in this setting as it is the most precise model. All models struggle to find all nuclei, evidenced by the low recall measure.

2.3.2 Tracking

Manually identified nuclei positions served as test data to evaluate and compare methods. Expert annotators missed in this data, and will miss nuclei in frames due to nuclear dimming or occlusion. Tracking results arise from assuming expert level detections, i.e. that the detections are of the same quality of the annotations. Frame-to-frame accuracy is reported by the proportion of correct matches to total nuclei in the subsequent frame. Longer term frame-to-frame tracking is prone to errors, and thus tracking is currently assumed to be done in a semi-automated fashion in which a trained user edits and verifies tracks in successive frames.

The data as presented features many frames with missing nuclei. The most challenging of scenarios occurs when two sequential frames have nuclei not present in each other. The augmented LAP is able to account for these cases by matching candidates from each set to nothing. However, the methodology is not designed for highly variant movement *within* a frame [34]. Nuclear movement introduced by untwisting error, worm elongation, and general nuclear movement are exacerbated by inaccurate detections. The GNN LP (Eq 2.9) was applied at four varying μm gates for nuclear movement: $10\mu m$, $15\mu m$, $20\mu m$, $100\mu m$.

Nuclear movement beyond each interval is a signal that a nucleus has dimmed in the subsequent frame while a newly emerging detection outside the cutoff is labelled a false positive.

Table 2.2 reports median matching accuracies for all gates over the test set. The gated LAP median accuracies converge to the non-gated LAP accuracy as the gate values increase. Varying the μm cutoffs cause significant changes in the median accuracy due to highly variant movement between frames. Certain frame show low nuclear movement while others show larger elongation of the worm. The elongation *stretches* coordinates along the body of the worm. Often nuclei move large distances, but they do so together. A checked detection set with an high cutoff would then allow the gated GNN LP to match nuclei despite highly variant bouts of movement.

	Median	IQR
GNN	0.952	0.120
GNN - 10 μm	0.615	0.529
GNN - 15 μm	0.702	0.564
GNN - 20 μm	0.708	0.610
GNN - 100 μm	0.952	0.145
Murty - K 5 - 10 μm	0.609	0.505
Murty - K 5 - 15 μm	0.690	0.569
Murty - K 5 - 20 μm	0.708	0.604
Murty - K 5 - 100 μm	0.952	0.134
Murty - K 30 - 10 μm	0.609	0.505
Murty - K 30 - 15 μm	0.701	0.554
Murty - K 30 - 20 μm	0.704	0.597
Murty - K 30 - 100 μm	0.952	0.135

Table 2.2: **Simple tracking approaches effectively track remapped nuclei when detections are perfect.** The augmented LAP allows for flexibility in handling imperfect detections. However, a cardinal assumption of the method is that nuclear movement is stationary [34]. This assumption does not hold due to the error injected from the untwisting process. A semi-automated approach in which a user corrects detections enables the *GNN* model effectively match nuclei in the straightened space.

2.4 Discussion

Tracking in the remapped coordinate space using a simple *GNN* tracker accurately tracks the majority of cell nuclei, but requires perfect annotations to be effective. The sources of nuclear movement are challenging to model accurately enough for reliable tracking despite the majority of frame-to-frame displacement being explained by embryo repositioning. The high variance in frame-to-frame displacement arises from error introduced by the coordinate remapping process. The body wall approximation via splines introduces systemic misplacement in some positions.

The burden thus ultimately falls on the detection process. The FCNs discussed in this work achieved results that correctly identified most of the nuclei, but are far from reliable in a fully automatic *GNN* tracker. The results point to a semi-automated solution. First image volumes are processed to detect nuclei. A trained user then remove debris, splits touching nuclei, and adds annotations for missed nuclei. The *GNN* tracker works in realtime, allowing for an refining approach to tracking nuclei. An initial pass will tend to correctly identify all nuclei from the prior frame. However, simply correcting any errors and rerunning while anchoring the correct predictions iteratively will produce correct frame-to-frame associations. This process applied recursively throughout an imaged worm drastically reduces both the manual effort and the time spent to track all densely labelled nuclei in a strain.

2.5 Future Work

Accurate nuclei detection stands as the foremost barrier to automated cell tracking. The semi-automatic interface and recursive cell identification algorithm mitigate the manual burden despite imperfect detections. The untwisting process accounts for cellular displacement attributable to embryo movement. However, the method is inadequate at times due to sharp bends or twists between pairs of seam cells. The residual cellular displacement is then a combination of inter-embryo movement and incorrectly mitigated displacement from untwisting. Bouts of elongation or warping complicate gating as nuclei may move minimally within the embryo or all in large coherent shifts. This variance furthers the need for accurate detections.

2.6 Acknowledgements

Post-Baccalaureate research fellows Gabi Schwartz and Daniel Del Toro-Pedrosa were assisted by testing and giving feedback on the tracking interface. Alexandra Bokinksy was instrumental for the integration of the tracking backend into the MIPAV Untwisting plugin. This research is supported by the Intramural Research Program within National Institute of Biomedical Imaging and Bioengineering at the National Institutes of Health. Andrew Lauziere's contribution to this research was supported in part by NSF award DGE-1632976. This work used the computational resources of the NIH HPC Biowulf cluster. (http://hpc.nih.gov).

Chapter 2: Multiple Hypothesis Hypergraph Tracking for Posture Identification in Embryonic *Caenorhabditis elegans*

2.1 Introduction

EHGM more accurately identified posture using hypergraphical models than established methods (Ch. 3.2), demonstrating the efficacy of interdependent modeling of seam cell movement. However, even top performing models often failed to discriminate between competing hypotheses. Posture identification in the low temporal resolution data was challenging due to the complete repositioning between images. The five minute recovery time between images was necessary to preserve the health of both the fluorophores and the embryo given the high spatial resolution necessary to identify other cells.

The 4D atlas project is broadly interested in tissue development across the embryo. Among the most studied is the nervous system. The development and functional capacity of the nerve ring, comprising just 302 cells, is widely studied as a model of human brain development. Nervous system development peaks between twitching and hatching as the bumbling embryo emerges from its eggshell demonstrating controlled motion. Late-stage twitching challenges behavioral analysis as observing the embryo using fluorescence microscopy suffers from the same tradeoffs present in the 4D atlas project. A detailed

behavioral analysis required rapid imaging to observe the continuously moving embryo. However, imaging with a high capture rate necessitates a low light dose to preserve both embryo and fluorophore health.

Late-stage embryos were imaged at 3 Hz (3 volumetric images per second) over 5 hours, producing ≈54000 image volumes. The increased capture speed entailed correlation between images. Posture identification at frame t is substantially easier in a qualitative sense (i.e. visually) if one knew the posture at frame *t-1*. Previous frame posture was shown to predict current frame posture in 27% of cases (Ch. 3.2) when using a graphical model. Here we investigated the integration of hypergraphical models to tracking posture. The task was modeled as a multiple object tracking (MOT) problem (Ch. 2.2.2) as opposed to a general object identification task in Ch. 3.

Recall the discussion on MOT in Ch. 2.2.2, particularly note the progression in model capacity from the global nearest neighbor (GNN) to the multiple hypothesis tracking (MHT) and application of Kalman Filtering. Frame-to-frame methods, such as the GNN, are prone to failure when faced with these obstacles. The optimization task is framed as a linear program (Eq. 2.9. Tracks and detections are paired by minimizing the summed displacement between points (Eq. 2.8).

MHT uses a "deferred decision" logic to make association decisions. Future detection sets are used in a probabilistic framework to disambiguate these challenging track-to-detection associations. The method requires a protocol to generate alternate association hypotheses. Murty's algorithm is the foremost method to generate the K leading solutions to the GNN in polynomial time [54, 55]. Each association hypothesis is then a feasible full track-detection update; future detections help to delineate comparable hypotheses by

individually computing likelihoods for each object. MHT can be expressed as a *multidimensional assignment problem* (MAP) [17, 30]. The MAP extends the traditional LAP to more than two frames, i.e. more than two successive point-sets [60]. Tracks at the previous frame are matched to detections in the sequential frame as a function of how those matches fit the expected object states (detections or expected locations if using a dynamical model). Detections from future frames are used to propagate tracks recursively; the matchings are subject to assignment problem constraints. The traditional MAP assumes a linear cost structure between objects, just as the MHT is expressed as well. The progression from GNN to MHT is illustrated in the second row of Fig. 2.1. Track update decisions in the second frame are made according to how decisions affect future associations. The final frame (Fig. 2.1-B) shows how a sudden motion can lead to incorrect associations.

The standard linear objective (as in Eq 2.9) explicitly models associations independently, i.e. the objective states that the association of track i to detection j is done irrespective of i' to j'. While this assumption enables efficiently scaling algorithms and broad applicability, it limits the association method from using interdependencies to better describe the association problem. Graphs are abstract models that describe pair-wise relationships between objects. Vertices refer to objects themselves while edges link vertices together. Attributed vertices and edges within each the track set and detection set can then be used to describe a correspondence. For example, the distance between vertices i and and i' can be used when considering a potential pair-wise match to detections j and j'. See edges between points in the third row of Fig. 2.1 for an example of a graphical representation. The correspondence problem is known as *graph matching*.

Hypergraphs extend the definition of a graph to include hyperedges which can specify

relationships among an arbitrary number of vertices. Hypergraph matching then concerns finding an optimal vertex correspondence between pairs of attributed hypergraphs. For example, a degree three hyperedge specifies a relationship between vertices i, i', i'' in the track hypergraph and vertices j, j', j'' in the detection hypergraph. Similar to the edges between points described above, an angle is visible between the red, blue, and green points in Fig. 2.1-C. Hyperedges are able to express higher degree joint relationships while edges can only express a bivariate relationship. Jointly, edges between pairs of points and angles between triplets of points assist in ensuring correct association decisions.

Multiple hypothesis hypergraph tracking integrates the deferred decision logic of MHT with the intricate modeling capability of hypergraphs. Hypergraphical models strengthen the association step of MOT by considering an intricate representation of simultaneous track to detection pairings. Data can be used to fit the hypergraphical model, further tuning the method to a particular application. Fig. 2.1 depicts the progression in tracking paradigm complexity from GNN to MHHT.

2.1.1 Related Research

Research on MOT in fluorescence microscopy has focused on single particle tracking (SPT). Particles defined as homogeneous independently moving objects are tracked under the assumption that frame-to-frame movement undergoes Brownian motion. Jaqaman et al. developed a two step approach to SPT [34]. The first step uses an expanded GNN cost matrix to specify cutoffs for both tracks with no associated measurements and measurements that are not associated to tracks, a program similar to Eq 2.9. The second

step solves a large LAP to handle linking the frame-to-frame associations, and both merge and split events throughout the image sequence. Padfield et al. combine both steps into one linear program [59]. A final image registration step is used to align successive frames with effort to clearly delineate large movements and object disappearances. The matching problem which Jaqaman et al. solved via the Hungarian algorithm is converted to a shortest path optimization problem and solved as a general linear program.

MHT based paradigms use future frame measurements, yielding a more computationally intensive but higher capacity MOT approach. Feng et al. express MHT as a multidimensional assignment problem [30]. The optimization objective is then rewritten into a linear program and solved as a binary linear program. The method is compared to Jaqaman et al [34]; MHT is more robust to noisy detections. Chenouard et al. present a probabilistic MHT framework. The MAP is again cast as a linear program to identify the optimal measurement to track association given measurements from a specified number of future frames. The program is solved via a branch-and-bound approach of the Simplex algorithm [17]. On the other hand, Rezatofighi et al. approach joint probabilistic data association [31](JPDA) within an MHT paradigm. The authors propose an algorithm to rank solutions to the LAP, producing a similar output to Murty's algorithm [66]. The ranked solutions serve as an approximation to the JPDA association costs. The process is then integrated into an MHT scheme to robustly associate measurements to tracks by using the JPDA soft assignment criteria and the deferred logic of MHT.

More recent methods to solve data association are inspired by the surge in popularity of neural networks. Spilger et al. demonstrate how recurrent neural networks using long-short term memory gates (LSTM) can be used to perform data association [75]. The LSTM

model learns assignment probabilities which then allow for the Hungarian algorithm to output a set of associations. Another recent approach uses both an LSTM and CNN for data association. Yao et al. use both networks with handcrafted and learned features to better inform the data association task [92]. The networks are used in tandem to *learn* association costs, similar to the approach of Spilger et al. [75]. Neural networks are used for expressive power to better contextualize the association step.

Euclidean distance is the underlying metric used by particle association methods [17, 34, 66]. The progression from traditional methods to recently published research is hallmarked by higher capacity models which are able to handle ambiguous association scenarios [75, 92]. MHHT builds upon the traditional MHT, but uses hypergraphical representations of object interdependencies to address challenging association decisions. Recent deep learning based approaches and proposed MHHT use data to more effectively perform data association.

2.1.2 Overview of MHHT & Application to *C. elegans*

MHT's adaptability and intuitiveness has led to its success on MOT tasks across domains. However, challenging MOT tasks which feature interdependent object motion could better be addressed by modeling these relationships. We developed multiple hypothesis hypergraph tracking (MHHT) to flexibly integrate interdependent association models into the prominent MHT paradigm. MHHT uses graphical and hypergraphical models to contextualize relationships between objects between points. While a graphical model uses *edges* to specify relationships between pairs of points, *hyperedges* allow for connecting an

arbitrary number of points. Graphical and hypergraphical relationships have been shown to improve the point-set matching process [27, 42, 96]. Point-set matching using such complex models is known to drastically inflate computational burden [42, 70]. However, the intricacy of the graph or hypergraph does not severely hamper computation as Murty's method is carried forward from MHT to generate hypotheses [23, 54, 55]. Each hypothesis constitutes a feasible full track update which is then evaluated using the posited hypergraphical model. MHHT uses Murty's algorithm in two distinct ways. The first explicitly calculates the K best hypotheses of each hypothesis at the preceding level, i.e. generating K^l hypotheses at level l in the search tree. Hypergraphical evaluations are then conducted in a depth-first search manner to identify the minimum cost path in the tree. The second applied an optimized version of Murty's algorithm [23, 54] to generate the K best hypotheses from the K preceding hypotheses at the prior level. Results presented above were achieved using the explicit search method.

The hypergraphical model f yields a cost when input an association between tracks $\mathbf{Z}^{(t-1)}$ and intermediate hypotheses formed at time steps $t, t+1, \ldots, t+N-1$ using detections $\mathbf{O}^{(t)}, \mathbf{O}^{(t+1)}, \ldots, \mathbf{O}^{(t+N-1)}$ given the hypergraph parameters $\mathbf{Z}$. Denote the cost $f^{(t-1,t)}$, then the hypergraphical objective between successive frames *t-1* and *t*. Then, the N frame MHHT objective for the association decision at time t can be expressed as a sum over $l = 0, 1, \ldots, N-1$: $\sum_{l=0}^{N-1} f^{(t+l-1,t+l)}$. The method enhances the data association step of MHT within a commonly used framework to mitigate challenges associated MOT.

MHHT was inspired by the nematode *Caenorhabditis elegans* (*C. elegans*), a small, free-living roundworm often studied as a model for neurodevelopment [88, 89]. The embryonic worm has a set of twenty skin cells, termed *seam cells*, which act as a "motion capture suit,"

revealing the coiled embryo's posture as it maneuvers in the eggshell. The cells run in pairs along the left and right sides of the embryo and are named in pairs: H0, H1, H2, V1, ..., V6, T, i.e. H0L and H0R comprise the H0 pair. Tracking the seam cells together constitutes recovering the coiled embryo's posture. Studying the embryo's movement throughout late-stage development yields insight into how the nervous system assumes control prior to hatching. Random bouts of muscular *twitching* in late-stage development complicate posture tracking. A custom cell nucleus detection model is applied to process all ≈ 54000 image volumes prior to tracking (Fig. 2.2-A). MHHT is applied using both data-independent and data driven hypergraphical models to perform posture tracking. The best performing models used graphical and hypergraphical relationships (Fig. 2.2-B,C) to more accurately model posture. Interdependent modeling improves performance on posture tracking on images from a held-out test embryo, demonstrating the effectiveness of the method on a cutting-edge task in computational biology.

2.2 Results

Seam cells expressing nuclear localized green fluorescent protein (GFP) were imaged by light-sheet fluorescence microscopy, which allows rapid optical sectioning with minimal photodamage [40, 90]. We recorded volumes at 3 Hz for more than 4.5 hours. Each image has $.65\mu m^2$ resolution with a 1.2 μm axial step size. Imaging in this manner did not appear to affect development and was not accompanied by detectable photobleaching. Consequently, we conclude that this imaging protocol did not interfere with normal embryonic development.

To track posture, seam cell nuclei must first be accurately detected (i.e. distinguished from each other and background). Using image volumes with manually annotated seam cells, we compared the performance of several image segmentation methods. The best performance was obtained using a 3D convolutional neural network [69, 98]. After detection, we evaluated methods for enabling comprehensive tracking of nuclear locations across all image volumes. Then, GNN and MHT approaches were compared to proposed MHHT models on posture tracking on a range of detection qualities: annotations, 3D U-Net, and IFT-Watershed [29, 49]. The three methods are vary by quality of detections; the first (annotations) assuming seam cell nuclei are always detected. The 3D U-Net was the highest performing method, automatically fit on a corpus of training data. The IFT-Watershed achieved the worst performance; it was manually tuned across training image volumes.

Biological insight into *C. elegans* inspired concepts of physical models to contextualize embryonic movement. The models themselves are expressed as graphs or hypergraphs. The embryo graph $G = (V, E)$ specifies a set of vertices V representing seam cells and edges E connecting seam cells locally. Edges appear posterior to anterior, laterally between pairs of nuclei, and diagonally between sequential pairs. The graph G serves as the basis for graphical and hypergraphical posture tracking models. Fig. 2.3 depicts a the embryo graph of an uncoiled embryo. The first graphical association model, denoted *Embryo*, compares changes in edge lengths frame-to-frame. Differences in the lengths of edges contribute to the cost of the track update. Annotated data were used to estimate statistics of a parametric model to further describe embryonic behavior. Two such models, *Posture* and *Movement*, were explored to track posture. *Posture* is the data-driven enhancement of *Embryo*; the hypergraphical model measures the consistency in the shape of the embryo throughout

successive frames. *Movement* is then a data enhanced version of the GNN, a graphical model evaluating patterned movement between nuclei.

2.2.1 Detection

Seam cell nuclei segmentation methods were evaluated by comparing each set of detections to annotated nuclei center points at each frame. Manually labelled annotation volumes were used to generate ground truth detection sets. Voxels corresponding to nuclei were labelled 1 while others are labelled 0. Each independent connected component represents a single nucleus. Each volume contains the nuclei visible to the annotator; ideally there are 19 nuclei in images from the first 3.5 hours of post-twitch development, and 21 when the Q neuroblasts split from the $V5$ pair. The tail pair nuclei appeared close due to limited spatial resolution, and are treated as one nucleus. Some nuclei were too dim to locate, in which case they were not annotated.

A total of 230 3D image volumes across three embryos were manually labelled using ImageJ [74]. The binary volumes were then randomly split into training, validation, and test sets of proportions 60%, 20%, and 20%, respectively.

Each proposed detection method ultimately yielded a set of detected objects. A linear program was used to match detections to annotations at each frame of the annotation set. One-to-one matching aligned annotations and detections per volume across the test set. The average precision, recall, and F1 score across test set volumes are reported for each method in Table 2.1.

Traditional methods for blob detection such as Watershed [29] and Laplacian of

Model	Precision	Recall	F1
IFT-Watershed [29]	0.81	0.80	0.80
LoG-GSF [46, 51]	0.95	0.90	0.92
Wavelet [58]	0.88	0.86	0.87
Mask-RCNN [33]	0.93	0.89	0.91
3D U-Net Dice [69]	0.94	0.91	0.92
3D U-Net Dice/BCE [98]	0.95	0.92	0.93
Stardist 3D [86]	0.91	0.88	0.89

Table 2.1: **3D CNNs outperform traditional segmentation methods for seam cell nuclei detection.** Nuclear detection results on a held out test set. Large kernel 3D U-Nets yielded both the highest precision and recall. However, even the 3D U-Nets could not accurately detect all seam cell nuclei.

Gaussians (LoG) [34] were compared to a variety of deep learning based approaches. Even recent methods such as *Stardist3D* [86] struggled to detect all seam cell nuclei. Most notable across methods is the disparity between precision and recall. While both traditional and deep learning based methods achieved comparable average precision, CNNs detected more of the dim nuclei. The latter result may be due to the CNNs' improved accuracies on voxels between close nuclei, more often returning disjoint nuclei instead of clumping multiple nuclei as detection. The adapted 3D U-Net trained on the summed Dice and BCE loss functions demonstrated the best results across the test set.

2.2.2 Tracking

Seam cell nuclei tracking stands in contrast to typical MOT tasks in fluorescence microscopy, and thus a broader metric was used to evaluate performance. An accurate representation of the posture requires accurate tracks of *all* seam cell nuclei. Incorrect associations violate the representation of the embryo and need to be corrected before cascading into further errors. The embryo also expresses several distinct behaviors throughout

late-stage development, such as relative stillness, fluid movement, and twitching. Two embryos were imaged and seam cell locations were then annotated across all (≈54000) volumes. The first imaged embryo was used to estimate MHHT model parameters, while performance was evaluated on the second annotated embryo, allowing for cross-validation of MHHT models. Test-set performance on the held out embryo gave insight into the generalizability of proposed models on future embryos. MHHT models were compared to two baseline methods: GNN (K=1, N=1) and MHT. MHT was conducted using MHHT with GNN costs between frames. Results presented were achieved by building explicit.

The predicted states were compared to the annotated ground-truth coordinates at each frame. A successful posture update will align one-to-one to the annotation points with each predicted nucleus state being sufficiently close to its respective annotation point, set at 7.5 μm. Frames which failed the test with either an incorrect association or losing track of at least one nucleus marked an error which would require expert intervention, and the coordinates were reset via a correction in the next frame. Methods were compared by their ability to maintain the embryonic posture across late-stage embryogenesis throughout variations in behavior. The percentage error rate is given by the ratio of frames in which a correction is required to all ≈ 54000 frames on the held out test embryo.

2.2.2.1 Annotations

A series of association models f were tested, increasing in complexity from the simple GNN (MHT, K=1,N=1) to the hybrid *Posture-Movement* model. Varying N from a single scan (N=1) up to N=5 highlighted the contribution of the deferred decision paradigm,

while varying $K = 1, 2, \ldots, 5$ allowed us to employ a wider search, improving the likelihood of encountering correct posture hypotheses.

Percentage error rates on the top quartile of movement across both pre-Q and post-Q frames are depicted in Table 2.2. The GNN achieved a baseline 6.04% error rate, while MHT (K=5, N=5) increases the error rate to 7.60%, demonstrating the inability of the linear association model to distinguish competing hypotheses. However, the progression from MHT to *Embryo* slightly reduced the error rate from GNN levels. The added discriminatory power of the graphical model allowed for identifying the correct posture more consistently. The data enhanced model *Posture* illustrates the contribution of annotations to tracking seam cell nuclei. *Posture-Movement* (*PM*) yields stronger results than either alone. The combined model reduces the error rate to 3.53%, a 42% reduction relative to the GNN error rate. Table 2.2(a) highlights the change in error rate across models with respect to the search width K, holding N=5 constant. Table 2.2(b) shows how parameterized shape based models *Posture* and *PM* particularly benefit from searching deeper, illustrating an improvement in performance with tree depth N.

2.2.2.2 Detections

Frames with debris and missed nuclei (i.e. imperfect detections) require evaluating more hypotheses (K) to find valid associations. Adapting MHHT to imperfect detections requires the introduction of gates $d_i^{(t)}$ with missing nucleus interpolation. Table 2.3 compares results by detection method, 3D U-Net vs. IFT-Watershed segmentation. Baseline methods are compared to MHHT models across quartiles of frame-to-frame movement, while varying

K	MHT	*Emb.*	*Post.*	*Move.*	*PM*
1	6.04	6.04	6.04	6.04	6.04
2	6.18	6.09	4.65	4.88	4.48
3	6.64	6.03	4.26	4.54	4.09
4	7.21	5.95	3.90	4.26	3.72
5	7.60	5.90	3.71	4.09	**3.53**

(a) Varying K, N=5.

N	MHT	*Emb.*	*Post.*	*Move.*	*PM*
1	6.04	5.37	4.24	4.04	3.97
2	6.98	5.53	4.07	4.06	3.88
3	7.26	5.77	3.87	4.08	3.61
4	7.41	5.90	3.82	4.04	3.58
5	7.60	5.90	3.71	4.09	**3.53**

(b) Varying N, K=5.

Table 2.2: **Data driven hypergraphical modeling better maintains posture than simpler methods.** Percentage error rates on frames in the top quartile of movement on the held out test embryo, assuming perfect detections. The combined *PM* model achieves an error rate of 3.53% (K=5, N=5), a 42% reduction in the error rate from the baseline GNN (6.04%). *PM* also benefits from increasing K and N, especially compared to simpler non-parametric models (MHT and *Embryo*). **Bolded** entries indicate the best in class (lowest) error rates.

gate size and detection method; MHT and MHHT methods use K=25 and N=2. Best in-quartile error rates are in bold; highlighting the effectiveness of data driven hypergraphical models in conjunction with state of the art detection methodology. The *PM* model achieved lower error rates across gate sizes and detection methods for each level of embryonic displacement, collectively lowering overall error by approximately 15% over baseline methods. Importantly, the MHHT models also demonstrated less variance within each quartile across gates for both detection methods. In particular, the 12.5 μm gate size posed a significant challenge for baseline methods, which were unable to distinguish between competing hypotheses.

Fig. 2.4 highlights relative error rates on the 12.5 μm gate across movement deciles as a function of search width K (5, 10, 25) with $N = 2$ on the 3D U-Net detections. *PM* sees a higher marginal reduction in error rate than MHT with respect to K. The improvements are attributed to the enhanced discriminatory power of the hypergraphical model over unary association methods. *PM* more effectively used computational resources allocated via the search width K than baseline MHT to perform posture identification.

	Gate (μm)	3D U-Net [98]				IFT-Watershed [29, 49]			
		GNN	MHT	*Embryo*	*PM*	GNN	MHT	*Embryo*	*PM*
Q1	2.5	3.44	2.94	2.78	2.82	3.80	5.09	3.53	3.68
	5.0	2.93	4.35	2.71	**2.57**	5.56	15.87	3.62	3.91
	7.5	3.32	5.10	2.73	2.62	14.30	22.65	6.94	6.62
	10.0	5.60	5.83	2.93	2.80	35.37	30.47	22.12	20.41
	12.5	7.09	6.71	3.67	3.34	49.07	40.89	39.38	37.31
Q2	2.5	6.00	5.12	4.72	4.75	6.68	7.38	5.87	6.10
	5.0	4.84	5.78	4.43	4.40	7.94	18.66	5.98	6.22
	7.5	4.93	6.70	4.52	**4.31**	17.94	25.17	9.48	9.08
	10.0	6.95	7.47	4.75	4.49	38.70	32.57	25.25	25.25
	12.5	8.78	8.28	5.24	5.00	52.41	43.82	41.87	40.13
Q3	2.5	10.90	8.40	8.03	7.86	11.70	11.58	9.78	9.67
	5.0	7.63	8.57	7.16	7.00	11.60	21.58	9.58	9.36
	7.5	7.80	9.69	7.17	**6.97**	20.93	28.85	13.04	12.33
	10.0	9.91	10.23	7.21	7.09	42.25	36.15	28.00	26.49
	12.5	11.85	10.67	7.71	7.55	56.08	47.63	46.16	44.69
Q4	2.5	22.68	19.96	18.70	18.69	23.86	22.95	21.71	21.46
	5.0	16.19	15.29	13.71	13.07	22.12	28.79	18.67	18.57
	7.5	14.35	15.81	12.67	12.19	29.32	36.20	21.13	20.95
	10.0	15.63	16.24	12.43	**11.99**	49.35	43.23	35.82	34.63
	12.5	17.35	16.71	12.84	12.37	62.18	54.52	52.31	51.19

Table 2.3: **MHHT outperforms baseline methods across levels of detection quality.** Comparing percentage error rates between baseline methods (GNN and MHT) and MHHT (*Embryo* and *PM*) when detections are imperfect. MHT and the MHHT models used K=25 and N=2. The progression in modeling capacity is compared to increasing gate size μm, embryonic movement (movement quartiles Q1-4) and detection method: 3D U-Net [98] vs. IFT-Watershed [29, 49]. MHHT achieved lower error rates across all degrees of movement and detection method. In particular, *PM* error rates were more robust to gate size, remaining accurate while GNN error rates increased. **Bolded** entries indicate best in class (lowest) error rates.

2.2.3 Eigen-embryos compactly describe posture and behavioural motifs

To describe embryo posture, we used seam cell positions to fit each side of the body with a natural cubic spline and then computed dorsoventral bending angles between adjacent seam cells (totalling 18 bend angles in each volume, Fig. 2.5-A,B). Four PCs captured approximately 88% of variation in the 18 angles (Fig. 2.5-C). The corresponding

eigenvectors of the four leading components (termed eigen-embryos) were stereotyped between animals. For example, PC1 captures ventral or dorsal coiling (i.e., all ventral or all dorsal body bends, respectively) while PC2 describes postures with opposing anterior and posterior bends. In this framework, embryonic posture can be approximated by a linear combination of eigen-embryos. The contribution of eigen-embryos shifted consistently across development, with less variance accounted for by PC1 as PC2 and PC3 gained prominence (Fig. 2.5-D). PC2 and PC3 approximate sinusoids with a phase difference of about 90° that can be combined to generate travelling waves of dorsoventral bending. The developmental shift towards PC2 and PC3 more closely approximates the adult motion pattern, where the top two PCs also describe sinusoids with a 90° phase shift [76]. The relatively limited dimensionality of dorsoventral bending observed in both adults and embryos is likely a consequence of muscle anatomy, with electrically coupled muscle bundles running ventrally and dorsally along the body [89].

The behavior of late-stage embryos exhibits several potential signatures of neuronal control (increasingly directed movement, dorsal coiling bias, and sinuous crawling). To establish a role for synaptic signalling, we analyzed *unc-13*(*s69*) mutants which have a nearly complete block in synaptic vesicle fusion and (consequently) profound movement defects [67]. In late-stage embryos (750 mpf), *unc-13* mutant movement was strongly impaired, as indicated by shorter seam cell trajectories and smaller diffusion coefficients (Fig. 2.5-E). Although their motion was severely restricted, *unc-13* mutants continued subtle movements in place, suggesting that spontaneous muscle contractions persist even when synaptic transmission is blocked. By contrast, at 530 mpf seam cell diffusion coefficients for *unc-13* mutants were indistinguishable from controls (Fig. 2.5-F), implying that synaptic

transmission is not required for the behavior of immature embryos. See [2] for more results and further discussion of behavioral analyses.

2.3 Discussion

MHHT is proposed as an extension of traditional MHT to leverage complex data association functions to handle interdependent object motion in tracking applications. The method is demonstrated to be more robust to poor detections than baseline tracking paradigms while achieving lower error rates on when tracking posture in embryonic *C. elegans*. Simulations across models and hyperparameter configurations demonstrate the generalizability of MHHT to similar tasks. The explicit tree search method adds discriminatory power at the cost of computation, particularly when increasing tree depth N. On the other hand, the more typically applied version of Murty's algorithm [23, 54] maintains K hypotheses at each scan. As such, the computational burden is linear in both K and N. Further simulations will need to be done using this method to better understand performance trade offs.

EHGM and associated posture identification models *Sides*, *Pairs*, *Posture*, and *Full* stood as a first look at modeling seam cell interdependencies. The hypergraphical models comprised intricate features in effort to overcome the little temporal context available in identifying posture. EHGM models stand in contrast to the best performing posture tracking model, *PM*. Presented MHHT models were much *simpler* in their constitutions. Cell-to-cell and edge-to-edge variations were used to quantify embryo movements at the 3 Hz capture rate. EHGM models were applied in the MHHT paradigm but yielded

poor results at all levels of detection quality. Recall the bias-variance tradeoff (Ch. 1.4); posture identification benefited from the more intricate models, but the task itself was more *challenging*. Posture tracking improved with relatively simpler models (PM), but failed with the EHGM models.

Hypergraphical optimization strategies differed between EHGM and MHHT. EHGM searches the entire permutation space for the optimal *hypergraphical match*. The exactness of EHGM ensure the identification of a globally optimal solution. On the other hand, MHHT orders the permutation space according to a linear model. Finding the optimal linear match across the permutation space is completed in polynomial time [35]. Like EHGM, MHHT enables both data-driven and data-independent approaches to hypergraphical modeling. In fact, the models applied to posture tracking are the data-driven versions of the data-independent models; *Movement* is a graphical model estimating correlated movement between cells, i.e. it is a data-driven GNN. Then, *Posture* estimates correlations between edge variations; it is a data-driven form of the graphical *Embryo* model (Chapter 2.5.2.3). Graphical and hypergraphical modeling functions are proposed in a context-free manner. While EHGM models were highly specified to posture identification, MHHT models are readily applicable in data-independent and data-driven manners to any MOT problem featuring interdependent motion.

Posture tracking in twitching embryonic *C. elegans* presents itself as an emerging problem archetype in MOT. Cutting edge microscopes allow for the observation of previously intimate of biological detail. MHHT improves upon tracking outcomes across detection capabilities, using annotated data to contextualize the complete embryo frame-to-frame. The method reduces the frame-wise error rate by approximately 11% over baseline methods

on the test embryo, correctly maintaining embryonic posture and thus eliminating the need for human intervention on over 700 additional volumes, out of approximately 54000 for an imaged embryo.

Our results suggest that embryos exhibit a stereotyped program for behavioral maturation in the final few hours before hatching, which comprises at least three phases (early flipping, an intermediate phase of reduced motility, and a late phase of mature motion). Following elongation, embryo behavior was initially dominated by flipping between all dorsal and all ventral body bends. Flipping behavior was not disrupted in *unc-13* mutants, implying that it does not depend on synaptic transmission. Flipping could be mediated by intrinsic oscillatory activity in muscles or by a form of neuronal signaling that persists in the absence of UNC-13 (e.g., gap junctions or an unconventional form of synaptic vesicle exocytosis). Because flipping comprises alternating all dorsal and all ventral bends, there must be some mechanism to produce anti-correlated ventral and dorsal muscle contractions.

Early flipping behavior is followed by a period of decreased motion. This slowdown is apparently not neuronally evoked, as it requires neither neuropeptide processing enzymes nor UNC-13. Slowing does coincide with a shift in the forces defining body morphology, from a squeeze generated by contraction of circumferential actin bundles of the epidermal cytoskeleton to containment within a tough, yet flexible extracellular cuticle [61]. This structural transition could impact behavior; however, preliminary experiments (not shown) suggest that reduced motion is a consequence of decreased muscle activity rather than a structural constraint limiting motion. A shift from cytoskeletal to exoskeletal control of body shape is likely reiterated at each larval molt and could contribute to molt-associated lethargus quiescence [62].

Following the inactive phase, late-stage embryos exhibit a mature pattern of motion. Mature motion comprises sinusoidal crawling, prolonged bouts of forward and reverse motion, and a rhythmic pattern of brief quiescent bouts. All of these features are grossly disrupted in *unc-13* mutants, implying that they are driven by synaptic circuits. Our prior study suggests that bouts of directed motion are mediated by the forward and reversal locomotion circuits that operate post-hatching [1].

2.4 Future Work

Similar to dense cell tracking (Ch. 2), posture tracking was burdened by segmentation errors. MHHT with the *PM* model reduced the overall error rate on the held-out test embryo by 29% over baseline methods (2.01% to 1.43%). However, the results were not as strong on the 3D U-Net detections; MHHT only reduced error by 11% (5.25% to 5.88%). Further feature engineering may yield a more accurate posture tracking model, but better nucleus detection is evidenced here to improve results.

2.5 Methods

2.5.1 Multiple Hypothesis Hypergraph Tracking

2.5.1.1 Hypergraphical Association

The established MHT paradigm [10, 22] relies upon a linear data association step to evaluate hypotheses. Successive associations across future frames ideally disambiguate the correct track update among sampled hypotheses. A hypergraphical model can better

evaluate the sampled hypotheses by accounting for interdependent object motion. An association function f measures the dissimilarity between established tracks $\mathbf{Z}^{(t-1)}$ and a hypothesized state update $\hat{\mathbf{Z}}^{(t)}$). The state estimate of the K sampled hypotheses $\hat{\mathbf{Z}}^{(t,k)}$ which minimizes the association function f is chosen as the joint state update, $\mathbf{Z}^{(t)}$. The method is most effective when a linear program can identify competitive state updates, but cannot accurately discern between candidates solely using the linear program costs.

A graph $G = (V, E)$ specifies a set of edges E connecting objects locally. Hypergraphical models allow for further evaluation of a sampled hypothesis under some further function of this edge set E. Hypotheses evaluated at time t: $\hat{\mathbf{g}}^{(t)} := \mathbf{g}(\hat{\mathbf{Z}}^{(t)}; E)$ describe the hypothesis according to predicted states at time t. Frame-to-frame differences in the attributed representations are assumed multivariate Gaussian: $(\hat{\mathbf{g}}^{(t)} - \mathbf{g}^{(t-1)}) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$. The Mahalanobis distance f is used to evaluate a hypothesized state representation:

$$f(\hat{\mathbf{g}}^{(t)}, \mathbf{g}^{(t-1)}; \hat{\mathbf{\Sigma}}^{-1}) = \sqrt{(\hat{\mathbf{g}}^{(t)} - \mathbf{g}^{(t-1)})' \hat{\mathbf{\Sigma}}^{-1} (\hat{\mathbf{g}}^{(t)} - \mathbf{g}^{(t-1)})}$$

The covariance matrix $\mathbf{\Sigma}$ is estimated from a corpus of annotated data. States of all n objects from frames $t = 1, 2, \ldots, T$ are used as pairs $\{(\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}), (\mathbf{Z}^{(2)}, \mathbf{Z}^{(3)}), \ldots, (\mathbf{Z}^{(T-1)}, \mathbf{Z}^{(T)})\}$ to estimate frame-to-frame variation in pairs of hyperedge differences: $\{(\mathbf{g}^{(1)}, \mathbf{g}^{(2)}), (\mathbf{g}^{(2)}, \mathbf{g}^{(3)}), \ldots, (\mathbf{g}^{(T-1)}, \mathbf{g}$ Define $\bar{\mathbf{g}} = \frac{\sum_{t=1}^{T} \mathbf{g}^{(t+1)} - \mathbf{g}^{(t)}}{T-1}$. Then, the covariance matrix is estimated:

$$\hat{\mathbf{\Sigma}} := \frac{1}{T-1} \sum_{t=1}^{T} (\mathbf{g}^{(t+1)} - \mathbf{g}^{(t)} - \bar{\mathbf{g}})(\mathbf{g}^{(t+1)} - \mathbf{g}^{(t)} - \bar{\mathbf{g}})'$$

2.5.1.2 Graphical Interpolation

Independently moving objects cannot provide insight into states of objects with missing measurements. However, specified dependencies between objects can be used to more precisely update tracks that do not receive a measurement at frame t. Our proposed graphical interpolation uses states $\mathbf{Z}^{(t-1)}$ and a hypothesis $\varphi^{(t,k)}$ with measurements $\mathbf{O}^{(t)}$ to complete the state update with predictions $\bar{\mathbf{z}}_i^{(t)}$, $i = 1, 2, \ldots, n$. The intermediate state update of hypothesis k: $\hat{\mathbf{Z}}(t, k)$ can then be written in terms of associated measurements $\mathbf{O}^{(t)}$ and predicted object positions $\bar{\mathbf{Z}}^{(t)}$:

$$\hat{\mathbf{z}}_i^{(t,k)} = \begin{cases} \bar{\mathbf{z}}_i^{(t,k)} & \varphi_k^{(t)} = 0 \\ \\ \mathbf{o}_j^{(t)} & \varphi_k^{(t)} = j \end{cases} \tag{2.1}$$

A graphical model specifying connectivity between objects serves as the basis for point interpolation. Each hypothesis $\varphi_k^{(t)}$ describes two disjoint sets of objects: objects which receive a measurement at t ($\varphi_k^{(t)} \neq 0$) and objects that do not receive a measurement at t ($\varphi_k^{(t)} = 0$). Designate the two disjoint sets $D^{(t,k)}$ and $U^{(t,k)}$, detected vertices and undetected vertices, respectively. Edges in which one element u is missing the other v is detected are used to predict the state of the missing vertex u. In the scenario in which for all edges connecting a missing vertex u, v is also a missing vertex the prior frame position is used for interpolation. Assume each undetected object $u \in U^{(t,k)}$ is a component of some edges in which the complementing vertex is detected: $\tilde{D}_u^{(t,k)} = \{v \in D^{(t,k)} : (u, v) \in E\} \subset U^{(t,k)}$, $n_u^{(t,k)} := |\tilde{D}_u^{(t,k)}|$. The graphically interpolated states are expressed:

$$\bar{\mathbf{z}}_u^{(t,k)} = \frac{\sum_{v \in \tilde{D}_u^{(t,k)}} [\hat{\mathbf{z}}_v^{(t,k)} - (\mathbf{z}_v^{(t-1)} - \mathbf{z}_u^{(t-1)})]}{n_u^{(t,k)}} \tag{2.2}$$

The estimated position of a missing object is the average of predicted positions under the edge set E. Inconsistencies in the complete state update will be penalized by hypergraphical association models. The method is extendable to hypergraphical or other forms of state interpolation involving object features derived from the image.

Fig. 2.6 illustrates an example of graphical interpolation applied to posture tracking. An anterior portion of Fig. 2.3 is depicted with H2L missing. Five nuclei give insight into H2L's position: H1L, H1R, H2R, V1L, and V1R. The prior volume points (blue) and the identified current volume points (red) yield intermediate predictions (green). The final prediction (purple) is an average of all five intermediate predictions. Encoded domain knowledge of *C. elegans* body structure via the graph is used to prune hypotheses which result in physically invalid postures.

2.5.1.3 Algorithm

MHHT adapts the established hypothesis oriented MHT proposed by Cox and Hingorani [22] to include graphical interpolation and further evaluation of the sampled hypotheses. In summary, Murty's algorithm identifies leading solutions to the data association problem proposed in Equation 2.9; states that are not directly updated undergo graphical interpolation when possible, and a hypergraphical association function f measures the cost of the complete state updates $\hat{\mathbf{Z}}^{(t,k)}$. This process is iterated recursively across multiple future frames to better inform the state update at time t. The parameter N dictates how many future

detection sets are considered. The deferred decision logic considers the expansion from states $\hat{\mathbf{Z}}^{(t,k)}$ to $\hat{\mathbf{Z}}^{(t+1,k)}$, $\hat{\mathbf{Z}}^{(t+2,k)}$, ..., $\hat{\mathbf{Z}}^{(t+N-1,k)}$ in the update at time t.

The recursion can be executed in two different ways. The first method forms an exponentially growing search tree in which paths from the initial state $\mathbf{Z}^{(t-1)}$ expand into K solutions. Each of the K solutions following interpolation yields a hypothesized state $\hat{\mathbf{Z}}^{(t,k)}$, initiating a recursion, adding to the cost of association at frame t. The search follows in a depth-first search manner with respect to time. The first complete posture sequence across N frames stands as the minimum cost hypothesis. The search continues with pruning to find the cost minimizing hypothesis accessible in the search tree. There are $\mathcal{O}(K^N)$ hypotheses are evaluated in the worst-case in the explicit tree search. Fig. 2.7-A depicts the explicit tree search method with K=2 and N=3. Murty's algorithm [54] is applied at each hypothesis to generate K hypotheses at the next time point. The cost-minimizing path is bolded, with the right hypothesis at N=1 being the chosen full-track update. Scissors indicate a pruned path due to an infeasible posture position. Darker colors (white to dark red) indicate an increasing cumulative cost as tracks are added to the search path.

The second method follows that of Miller, Stone and Cox [54] and Cox and Miller [23] in which the K best hypotheses of the K preceding hypotheses are found in one call of Murty's algorithm [55]. This second method is used in leading MHT applications [22]. The trade off is in losing the ability to compare *all* K^2 hypotheses from a previous subproblem as opposed to only the K leading hypotheses (according to the linear model). There are in total NK hypotheses evaluated using this method. Fig. 2.7-B shows the optimized version of Murty's algorithm typically applied in MHT [22] with K=3 and N=3. Again, the bolded

path identifies the cost-minimizing path. However, not all K^l hypotheses at level l (but for l=1) are evaluated using the hypergraphical model. As such, a suboptimal path may be returned in exchange for reduced computation.

The introduction of graphical modeling requires a fixed structure among tracked objects, with the n objects being set a priori. New objects require updating the graphical models for interpolation and association, and will be tracked until manually removed from the object set. MHHT is designed to track a set number of objects with specified interdependencies despite noisy detections and large bouts of coherent motion.

2.5.2 Posture Tracking in Embryonic *C. elegans*

2.5.2.1 Overview

Posture is defined as the complete state identification of *all* seam cells, which approximates the shape of the coiled embryo. Seam cell nuclei tracking is achieved via a *detect and track* paradigm. The image volumes are first processed in batch via a 3D convolutional neural network (CNN) to detect nuclei. The detections are used to track seam cells throughout the image sequence. Tracking is achieved via our proposed method: multiple hypothesis hypergraph tracking (MHHT).

2.5.2.2 Seam Cell Nuclei Detection

Accurate seam cell nuclei detection is especially challenging for two distinct reasons. First, nuclei frequently appear dim to the degree that an expert has to infer their positions from the more visible adjacent nuclei. The second issue arises due to the spatial resolution

of the image volumes. The imaged embryo will often depict two seam cell nuclei so close to each other that they appear as one larger nucleus. Fig. 2.8 depicts three XY maximum intensity projections from sequential image volumes. The shapes and intensities of fluorescently labelled seam cell nuclei vary throughout imaging. The tail nuclei are smaller than other nuclei and much closer than nuclei of other pairs. The limited spatial resolution results in the tail nuclei appearing merged as one nucleus. Red dots are placed on seam cell nuclei in the tail pair in each maximum intensity projection.

Seam cell nuclei detection is achieved by a convolutional neural network (CNN). A 3D U-Net style architecture is employed to perform semantic segmentation on the *C. elegans* embryo image volumes [69, 98]. Fig. 2.9 depicts the model architecture. The established 3D U-Net is augmented to use a size $(5, 5, 5)$ kernel, extending the field of view at each layer. Strided 3D convolution layers downsample by a factor of two across lateral dimensions while preserving axial resolution. The limited axial resolution encodes more information per planar image due to the explicit downsampling occurring during imaging. The number of filters in each layer doubles from 16 to $32, 64, 128$ when downsampling.

The loss function is a uniformly weighted average of the binary cross-entropy and negative dice coefficient. The cross-entropy portion prioritizes accurate prediction of challenging individual voxels between close nuclei, but may produce noisy predictions. On the other hand, the dice coefficient prioritizes structural similarity in clusters of voxels constituting nuclei, but is known to consolidate sigmoid outputs at extreme values 0 and 1, resulting in close nuclei being labelled as one supervoxel [73]. Denote $\mathbf{y} \in \{0, 1\}^{X \times Y \times Z}$ and $\hat{\mathbf{y}} \in [0, 1]^{X \times Y \times Z}$ as the binary ground truth and sigmoid output tensors respectively. The loss can then be written:

$$\mathcal{L}(\mathbf{y},\hat{\mathbf{y}}) = \underbrace{-\sum_{i=1}^{X}\sum_{j=1}^{Y}\sum_{k=1}^{Z}[y_{ijk}\log(\hat{y}_{ijk}) + (1-y_{ijk})(\log(1-\hat{y}_{ijk})]}_{\text{Cross Entropy}} + \underbrace{\frac{-2\sum_{i=1}^{X}\sum_{j=1}^{Y}\sum_{k=1}^{Z} y_{ijk}\hat{y}_{ijk}}{\sum_{i=1}^{X}\sum_{j=1}^{Y}\sum_{k=1}^{Z}(y_{ijk}^2+\hat{y}_{ijk}^2)}}_{\text{Dice Coefficient}}$$

The model is trained via the Adam optimizer with an initial learning rate of 0.00075 [37]. The resulting image volume contains values $\hat{y}_{ijk} \in [0,1]$. The output is thresholded and then independent connected components are returned as predicted instances of seam cell nuclei. The centroids of each supervoxel serve as the detection set for each image volume.

Imperfect detections due to low spatial resolution contribute to a challenging MOT task. Detection methods often yield debris and fail to detect all nuclei due to either dimness or clumping adjacent nuclei together. The proposed 3D U-Net [98] outperforms both traditional methods: Imaging Forest Transform Watershed [29], Laplacian of Gaussian [46, 51] and a Wavelet method [58], and modern CNN based approaches: Mask-RCNN [33], and Stardist 3D [86] in detecting seam cell nuclei. Descriptions and implementations of detection methods can be found in the appendix (section 2.2.1).

2.5.2.3 Posture Tracking

Posture tracking is achieved using MHHT with a proposed hybrid graphical and hypergraphical model. The explicit tree search method of Murty's algorithm is used to achieve the lowest cost hypotheses at expense of computation. Disjoint body segments are formed by sequential pairs of seam cells. Incorrect associations may yield a graphical

representation in which body segments intersect with each other. The Möller-Trombore algorithm is used to estimate intersection among body segments and prune invalid hypotheses [56].

Embryo & *Posture* The *Embryo* and *Posture* models penalizes associations which distort the embryo's shape. Accurate seam cell nuclei state updates will adhere to the physical confines of the embryo, even during sudden twitches. Edges illustrated in Fig. 2.3 vary in length as the states update frame to frame. Denote the edges $E = [e_1, e_2, \ldots, e_M]$, where edge $e_j = (u_j, v_j)$ describes a relationship between nuclei u_j and v_j. Each vector $\mathbf{e}_j^{(t)} = \mathbf{z}_{u_j}^{(t)} - \mathbf{z}_{v_j}^{(t)}$ describes the chord connecting states of nuclei u_j and v_j at time t. Then, the vector $\mathbf{E}^{(t)} = [\|\mathbf{e}_1^{(t)}\|_2, \|\mathbf{e}_2^{(t)}\|_2, \ldots, \|\mathbf{e}_M^{(t)}\|_2]$ describes lengths of these chords. Differences in chord lengths between frames $\mathbf{E}^{(t)}$ - $\mathbf{E}^{(t-1)} \in R^{M\times 1}$ form the basis of the association cost. Then, the *Embryo* model is defined:

$$f_E(\hat{\mathbf{Z}}^{(t)}, \mathbf{Z}^{(t-1)}) = \sqrt{(\hat{\mathbf{E}}^{(t)} - \mathbf{E}^{(t-1)})' \mathbf{I} (\hat{\mathbf{E}}^{(t)} - \mathbf{E}^{(t-1)})} = \sum_{j=1}^{M} \sqrt{(\hat{\mathbf{E}}^{(t)} - \mathbf{E}^{(t-1)})^2} \tag{2.3}$$

Annotated data are used to estimate covariances between the M differences across state updates. The resulting covariance matrix $\hat{\Sigma}_P$ scales differences in chord lengths among the M chords present in G:

$$f_P(\hat{\mathbf{Z}}^{(t)}, \mathbf{Z}^{(t-1)}; \hat{\mathbf{\Sigma}}_P^{-1}) = \sqrt{(\hat{\mathbf{E}}^{(t)} - \mathbf{E}^{(t-1)})' \hat{\mathbf{\Sigma}}_P^{-1} (\hat{\mathbf{E}}^{(t)} - \mathbf{E}^{(t-1)})} \tag{2.4}$$

Both *Embryo* and *Posture* are further characterized by the unary costs specified by

$\mathbf{C}$ (Equation 2.8) and the evaluated hypothesis $\varphi^{(t)}$: $\sum_{i=1}^{n} \mathbf{C}_{i,\varphi^{(t)}}$.

Movement *Movement* extends the traditional GNN cost to penalize *unnatural* movement between states. The distance between states at $t-1$ and t, $\sum_{i=1}^{n} \|\mathbf{z}_i^{(t)} - \mathbf{z}_i^{(t-1)}\|_2$, can be scaled by the inverse covariance matrix describing motion between pairs of nuclei. The states $\mathbf{z}_i^{(t)} = [x_i^{(t)}, y_i^{(t)}, z_i^{(t)}]$ and $\mathbf{z}_j^{(t-1)} = [x_j^{(t-1)}, y_j^{(t-1)}, z_j^{(t-1)}]$ can be expressed as element-wise differences:

$$\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)} = \begin{bmatrix} \mathbf{z}_1^{(t)} \\ \mathbf{z}_1^{(t)} \\ \vdots \\ \mathbf{z}_n^{(t)} \end{bmatrix} - \begin{bmatrix} \mathbf{z}_1^{(t-1)} \\ \mathbf{z}_2^{(t-1)} \\ \vdots \\ \mathbf{z}_n^{(t-1)} \end{bmatrix} = \begin{bmatrix} x_1^{(t)} \\ y_1^{(t)} \\ z_1^{(t)} \\ x_2^{(t)} \\ y_2^{(t)} \\ z_2^{(t)} \\ \vdots \\ x_n^{(t)} \\ y_n^{(t)} \\ z_n^{(t)} \end{bmatrix} - \begin{bmatrix} x_1^{(t-1)} \\ y_1^{(t-1)} \\ z_1^{(t-1)} \\ x_2^{(t-1)} \\ y_2^{(t-1)} \\ z_2^{(t-1)} \\ \vdots \\ x_n^{(t-1)} \\ y_n^{(t-1)} \\ z_n^{(t-1)} \end{bmatrix} = \begin{bmatrix} x_1^{(t)} - x_1^{(t-1)} \\ y_1^{(t)} - y_1^{(t-1)} \\ z_1^{(t)} - z_1^{(t-1)} \\ x_2^{(t)} - x_2^{(t-1)} \\ y_2^{(t)} - y_2^{(t-1)} \\ z_2^{(t)} - z_2^{(t-1)} \\ \vdots \\ x_n^{(t)} - x_n^{(t-1)} \\ y_n^{(t)} - y_n^{(t-1)} \\ z_n^{(t)} - z_n^{(t-1)} \end{bmatrix} \in R^{3n \times 1} \tag{2.5}$$

Each pair of nuclei has an estimable 3×3 covariance matrix specifying the relationship between movement along each axis. The resulting block $3n \times 3n$ covariance matrix, $\mathbf{\Sigma}_M$ then scales the difference between states:

$$f_M(\hat{\mathbf{Z}}^{(t)}, \mathbf{Z}^{(t-1)}; \hat{\mathbf{\Sigma}}_M^{-1}) = \sqrt{(\hat{\mathbf{Z}}^{(t)} - \mathbf{Z}^{(t-1)})' \hat{\mathbf{\Sigma}}_M^{-1} (\hat{\mathbf{Z}}^{(t)} - \mathbf{Z}^{(t-1)})} \tag{2.6}$$

The *Posture* and *Movement* models are combined additively to produce the *Posture-Movement* (*PM*) model.

2.5.2.4 An Interface for Posture Tracking

We developed our detect and track method with the goal of extracting seam cell coordinates over time, thereby allowing detailed analysis of embryonic behavior. The image volumes are processed in batch, yielding the complete detection set. An interface is developed in Python to perform posture tracking in embryonic *C. elegans* via *MHHT*. The web based interface is freely available: `https://github.com/lauziere/MHHT`. Track correction is performed in *MIPAV*: Medical Imaging, Processing, and Visualization [21, 53]. MIPAV is available: `https://mipav.cit.nih.gov/`.

2.5.2.5 Creating Annotations

Image volumes were processed in batch via the large kernel 3D U-Net in section 2.5.2.2. Tracking was achieved via a gated GNN. An overlaid graphical representation (Fig. 2.3) serves as visual cue for correctly updating posture. The user is prompted to verify if the posture is correct; the program generates a file structure that MIPAV can recognize and outputs the necessary parameters to view the data if the tracks need to be edited. The process is applied recursively frame-to-frame throughout the image sequence until hatching.

2.6 Acknowledgements

This research is supported by the Laboratory of High Resolution Optical Imaging within the National Institute of Biomedical Imaging and Bioengineering at the National Institutes of Health. This work used the computational resources of the NIH HPC Biowulf cluster. (http://hpc.nih.gov). We thank Mr. Brandon Harvey for providing annotation data and Dr. Ghadi Salem for early advice on nuclear segmentation strategies. Andrew Lauziere's contribution to this research was supported in part by NSF award DGE-1632976. The code and data that support these studies are available at `https://github.com/lauziere/MHHT`.

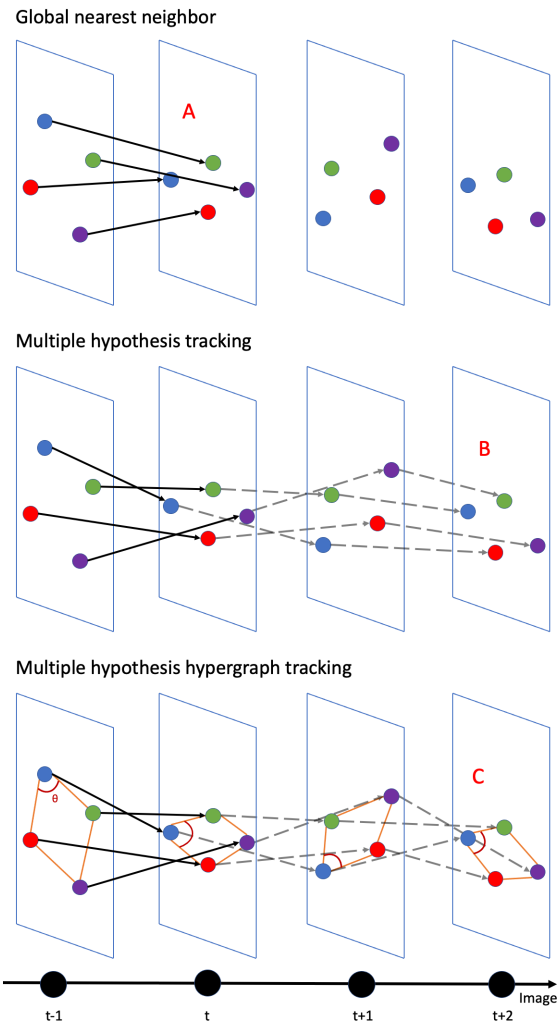

Figure 2.1: **Interdependent object tracking can be achieved via MHHT.** Four objects (blue, red, green, purple) are observed in a sequence of images: t-1, t, t+1, and t+2. The established tracks at frame t-1 are to be continued across the next frames. Three tracking paradigms are presented: global nearest neighbor (GNN, top), multiple hypothesis tracking (MHT, middle), and multiple hypothesis hypergraph tracking (MHHT, bottom). Frame-to-frame methods such as GNN can only use information from the sequential frame (solid black arrows) to update tracks. A sudden rotation of all objects (A) may cause GNN to misidentify objects. However, Multiple Hypothesis Tracking (middle) uses information from future frames t+1 and t+2 (dashed black arrows) to disambiguate tracking decisions at frame t. Object identities can be recovered under smooth motion, but sudden large movements (B) may introduce tracking error. Both GNN and MHT perform tracking by treating objects as independent entities. MHHT (bottom) augments MHT to allow interdependent modeling of object tracks (orange lines). Correlated object movements yield additional structure such as angles (dark red). Hypergraphical modeling adds richer context than MHT or GNN, enabling accurate tracking of objects with interdependent motion (C).

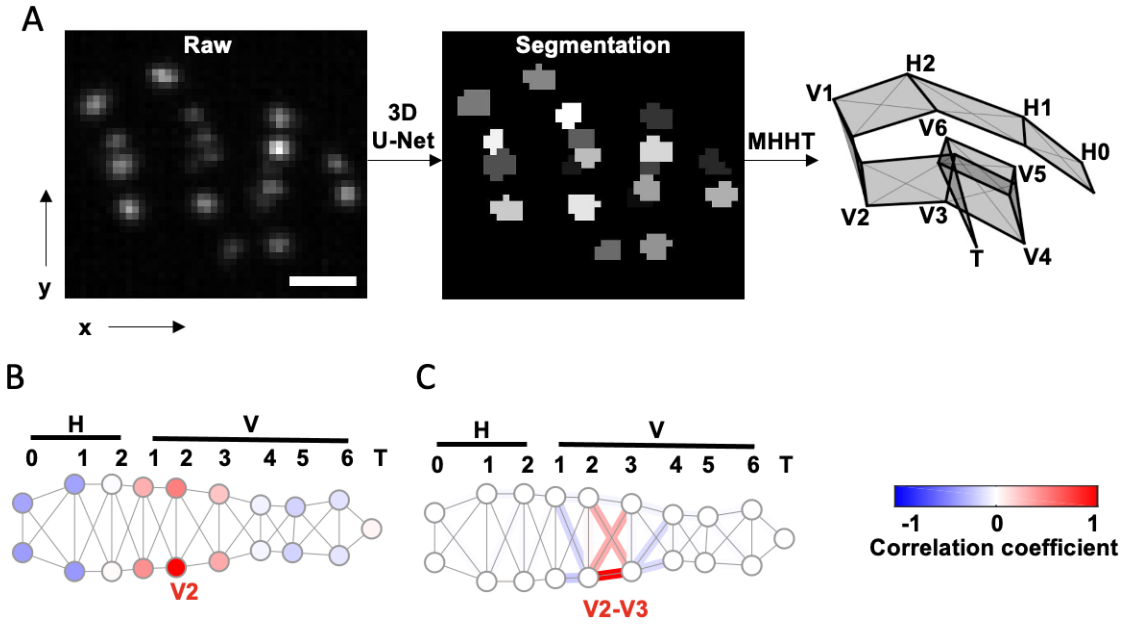

Figure 2.2: **Overview of MHHT applied to track posture in embryonic *C. elegans*.** A) Raw image volumes are processed via a custom segmentation model to find seam cell nuclei. Scale bar 10 μm. The detection centers are then used with prior frame tracks and future detection sets to recover posture, i.e. identify *all* seam cells. The process is repeated for the last hours of development, $\approx$ 54000 image volumes. B) A flattened representation of the seam cells. The cells of each pair anterior to posterior: H0, H1, ..., V5, V6, T are related to each locally other via a graph. The V2 cell's movement is positively correlated (red) with nearby cells while its movement is negatively correlated (blue) with anterior and posterior cells. C) A hypergraphical model relates edges to each other. Length variation in V2-V3 edge is positively correlated to variation of edges within that portion of the embryo while negatively correlated with edges adjacent to that portion of the embryo.

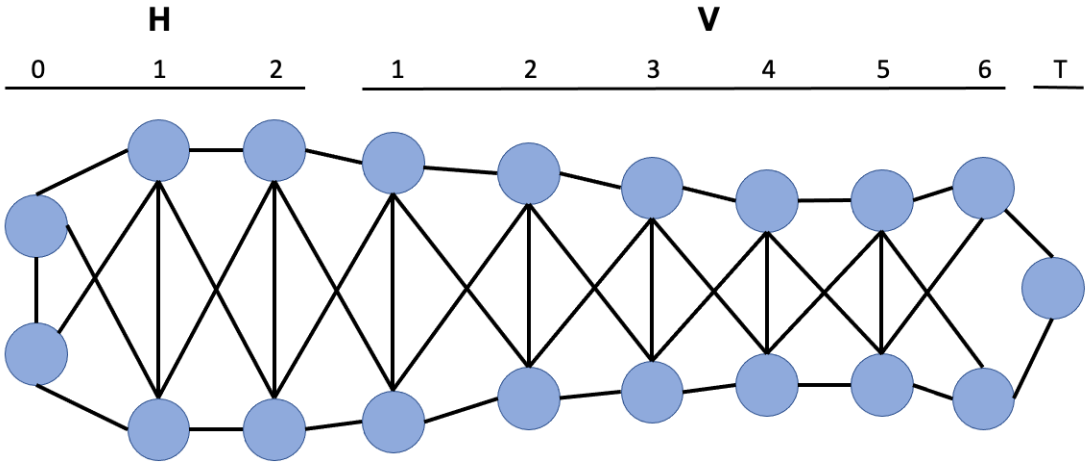

Figure 2.3: **The embryo graph describes interconnectivity among seam cells and serves as a basis for hypergraphical modeling.** Interdependence between seam cells is modeled via a graph. Pairs of cells are labelled anterior to posterior: H0, H1, H2, V1, ..., V5, V6, T. Physiologically close cells are linked via an edge. The graphical model underpins both the interpolation step and hypergraphical data association steps of the tracking process. Specifically, correlated movement between paired nuclei and correlated edge length variations are considered when quantifying a track update.

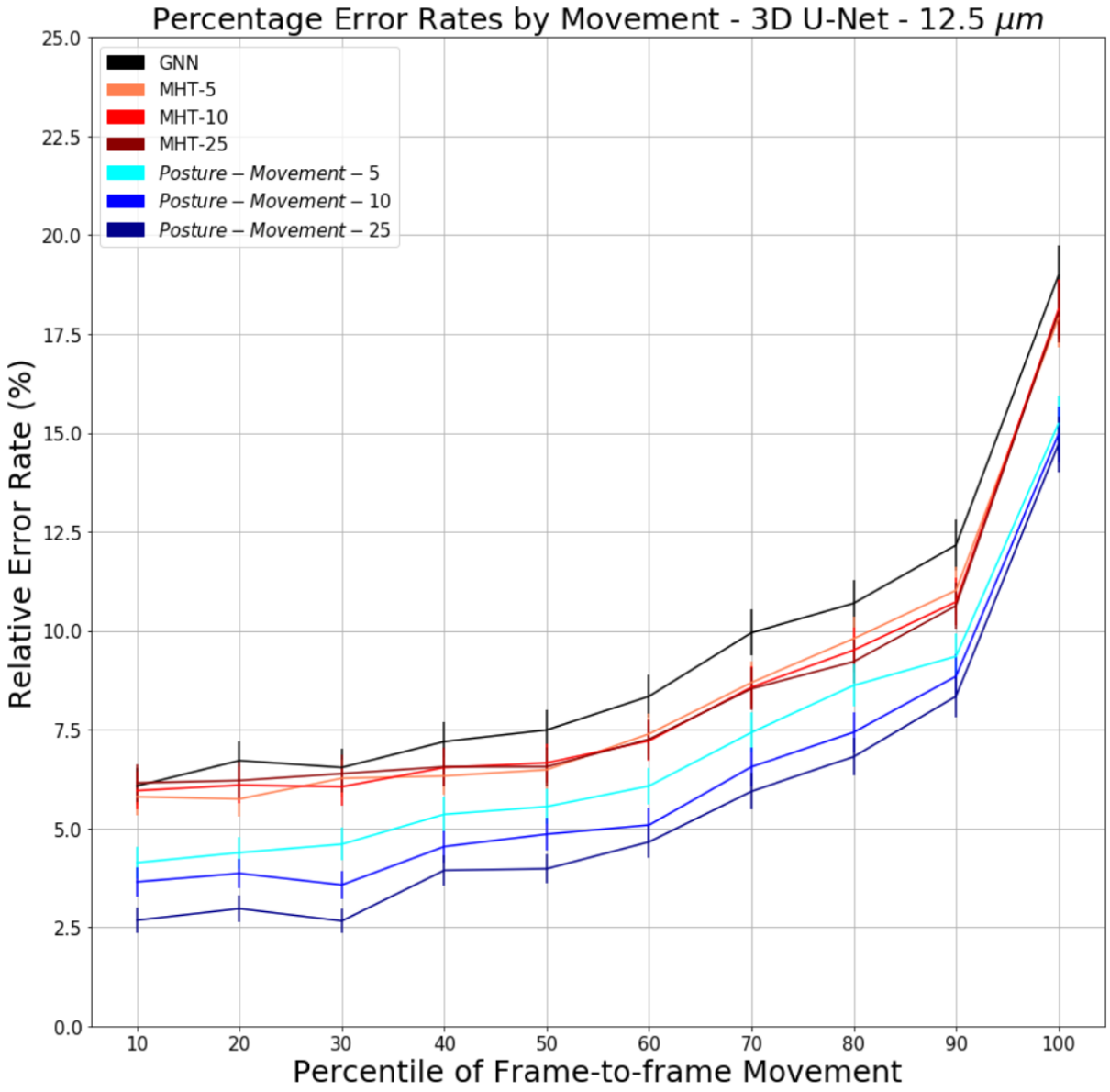

Figure 2.4: **Hypergraphical models improve performance on challenging association decisions.** Percentage relative error rates by movement decile, focusing on the 12.5μm gate. GNN, MHT (K=5, 10, 25. N=2), and PM (K=5,10,25. N=2) are compared. The MHT results are clustered while the PM results improved with increased K. The separation between PM models with varying K highlights the increased discriminatory power in the hypergraphical model.

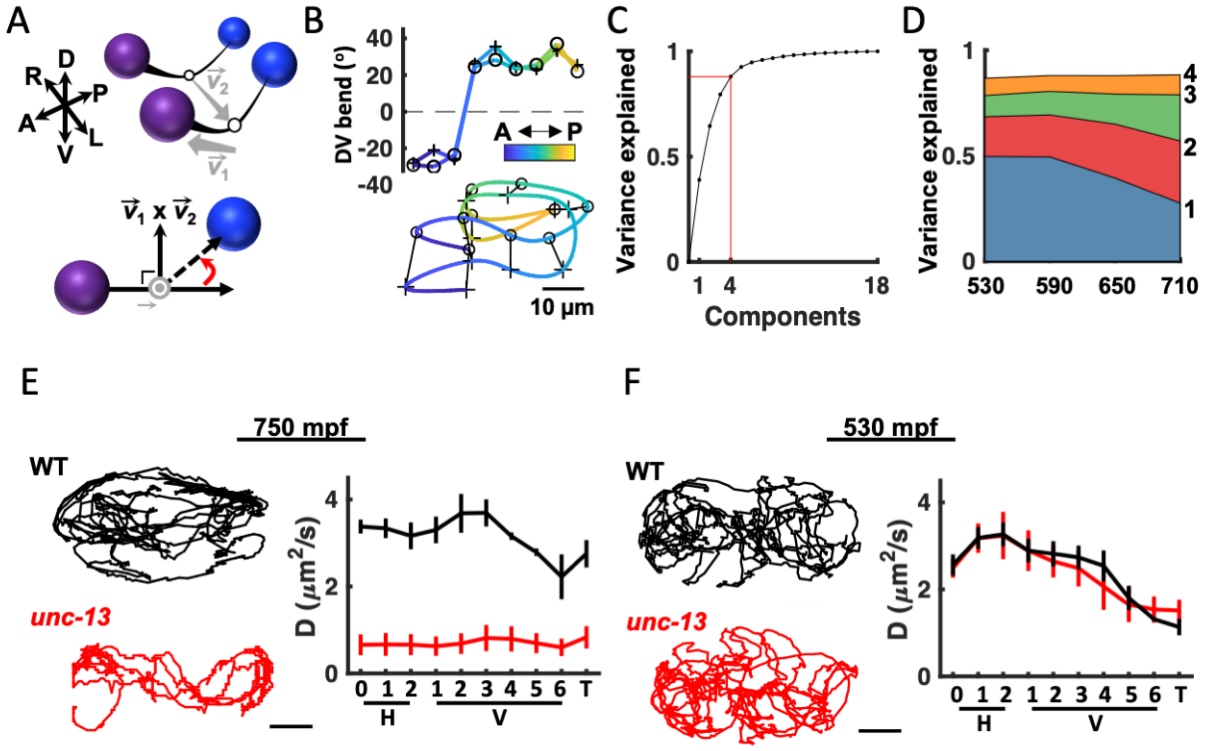

Figure 2.5: **Eigen-embryos compactly describe behavioral maturation.** A) A schematic illustrating how dorsoventral (DV) bends are defined. Top: Seam cell nuclei on each side of the body are fit with a natural cubic spline (black line). Vector $\vec{v}_2$ links the midpoints between adjacent seam cell nuclei (open circles). Bottom: View looking down $\vec{v}_2 2$ to highlight a DV bend angle on one side of the body (red arrow). B) DV bend angles between all adjacent seam cells are used to define an embryo's posture. DV bends along the left (+) and right (o) sides of an embryo (top) are plotted (top). The posture model for this embryo is shown (bottom). Position along the anteroposterior axis is indicated by the color gradient. C) The fraction of the total variance captured by reconstructing postures using 1 through all 18 principal components is plotted. D) Fraction of the total variance captured by the first 4 principal components is plotted as a function of embryo age (minutes post fertilization, mpf). E & F) ***unc-13* mutants have a late-stage motility defect.** E) Seam cell motions are compared in WT and *unc-13* mutant embryos at 750 mpf (E) and 530 mpf (F). Representative 10 minute trajectories for the H1 seam cell (left) and mean diffusion coefficients for all seam cell pairs (right; mean +/- SEM) are shown. Scale bar, 10 μm. Sample sizes, 750 mpf (3 WT, 3 *unc-13*); 530 mpf (3 WT, 2 *unc-13*).

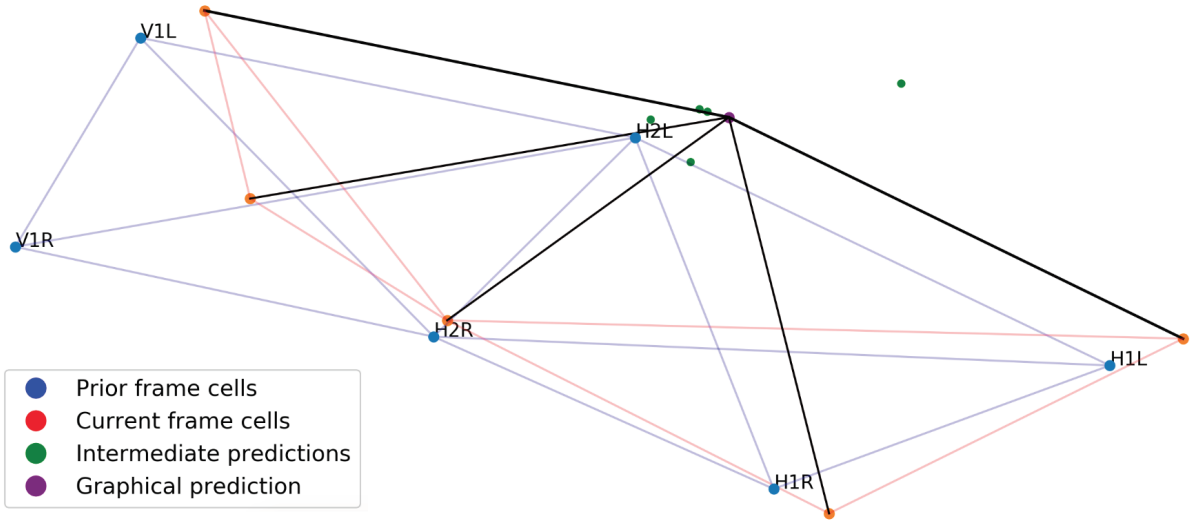

Figure 2.6: **Graphical interpolation leverages hypothesized associations to yield a more robust full track update.** Graphical interpolation is applied to posture tracking via the embryo graph. The blue graph represents an anterior part of the posture of the image volume show in Fig. 2.3. The red graph arises from a posture hypothesis in the next frame in which H2L is missing. Black edges comprising identified relevant nuclei are used in conjunction with the prior frame graph to predict the position of H2L. The green points are intermediate predictions with the purple point being the final prediction, an average of individually predicted black points via Eq 2.2.

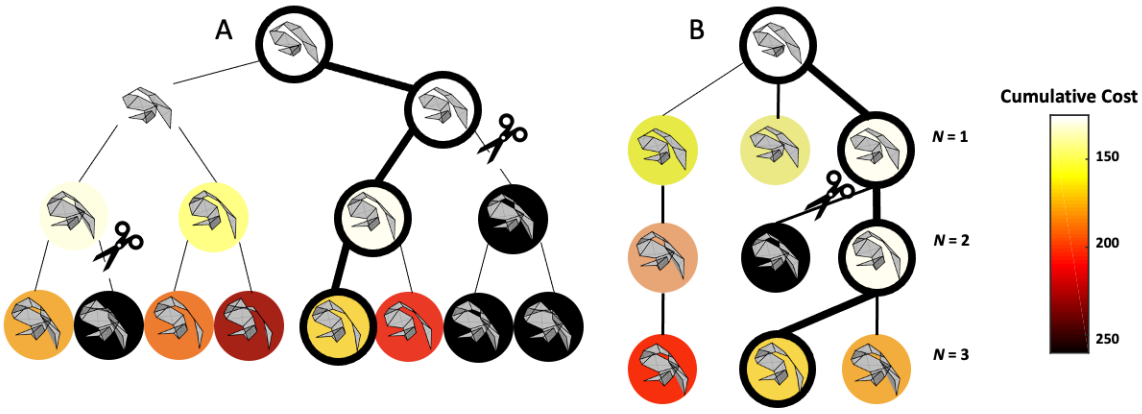

Figure 2.7: **MHHT search trees constructed using two different methods.** Murty's algorithm applied in two different manners to perform posture tracking. A) The traditional implementation of Murty's algorithm will generate the K best hypotheses from each preceding hypothesis. The exponentially growing search tree yields a more thorough exploration of hypotheses. On the other hand, a more typically used version of Murty's algorithm will generate the K best hypotheses from the K preceding hypotheses. Worst case computation is reduced from exponential in N to linear in N. Bolded circles highlight the cost-minimizing path; scissors denote a pruned hypothesis due to Murty's algorithm returning a biologically infeasible posture.

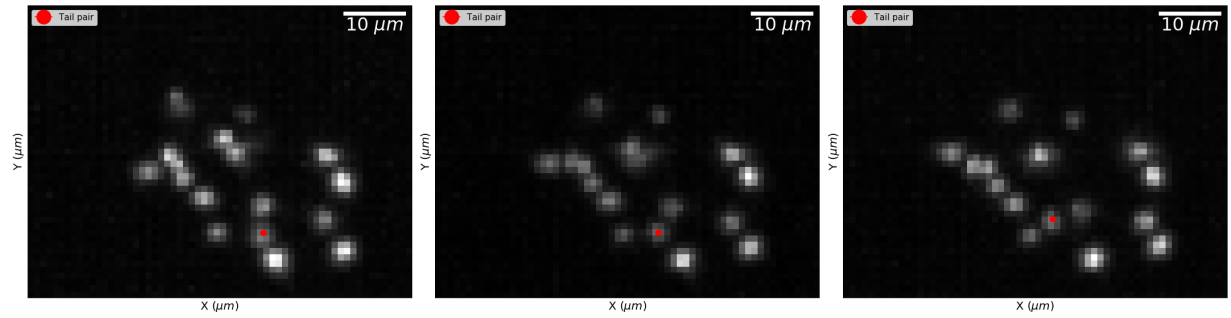

Figure 2.8: **Low spatial resolution, high temporal resolution volumetric imaging allows observation of late-stage behavior.**Three sequential XY maximum intensity projections of images captured at approximately 2 hours before hatching, imaged at 3 Hz. Low illumination dose permits long-term imaging of rapid movements in the embryo without obvious phototoxicity. The apparent size and intensity of seam cell nuclei fluctuate throughout imaging, but are largely homogeneous and are indistinguishable based on appearance alone. Red dots are placed on nuclei in the tail pair.

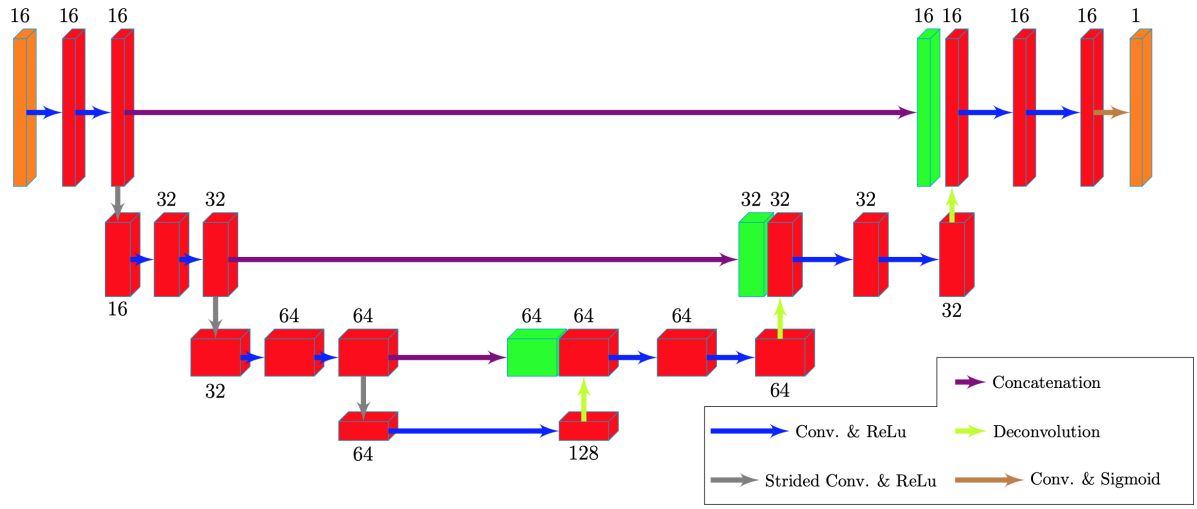

Figure 2.9: **A 3D U-Net is trained to perform semantic segmentation on image volumes.** The input (orange) is a full 3D image volume. Successive convolutions (blue arrows) and strided convolutions (gray arrows) encode information. Deconvolution layers parametrically upsample input layers towards the input size. The number of filters in each layer (1, 16, 32, 64, 128) is displayed. The output is (orange) is the same size as the input, each voxel value $y_{ijk} \in [0, 1]$ representing the likelihood of being part of a nucleus.

Chapter 2: Conclusions

Multidisciplinary research between natural scientists, optical physicists, and mathematicians stands to advance discovery of biological phenomena. In particular, questions concerning developmental biology often requires observation at the cellular level. Techniques such as electron microscopy and fluorescence microscopy (Chapter 1.3) enable such observation. However, the quantity and quality of subsequent image data increases as the capability of microscopes improves. The development and application of algorithms and data-driven statistical modeling is integral for efficiently analyzing image data.

C. elegans embryogenesis inspired mathematical research contributed in this thesis. Biologists and neuroscientists study the worm as a model for nervous system development. However, the onset of muscle cells approximately halfway through embryo-development complicates further observation of the embryo. The rapid twitching movements cause the embryo to move abruptly in its eggshell (Chapter 1.2); limitations in fluorescence microscopy impose tradeoffs when imaging the embryo (Chapter 1.3). Two research problems concerning *C. elegans* embryo-development were explored: completing the 4D atlas and behavioral analysis. Point-set identification problems have been largely solved using efficient matching algorithms or registration methods. Posture identification emerged as a problem unlike any other in computer vision. The *C. elegans* embryo's particular flexibility in

conjunction with its developmental elongation and fast twitching bouts come together to yield a uniquely challenging point-set identification task.

The 4D atlas prior to twitching has been completed [71]. Late-stage development encompasses the moments in which cells form complete tissues and organs, as well as when the embryo becomes capable of life outside the eggshell. Completion of the 4D atlas requires tracking cells throughout late-stage development despite bouts of twitching motion. The seam cells act as a "motion capture suit" to register sequential images; cells are then remapped according to an approximation of the embryonic posture (Figure 3.2) [21]. Joint identification of the seam cells defines the posture. Manual posture identification requires biological knowledge and is labor intensive. As such, posture identification is a significant barrier to completing the 4D atlas.

Behavioral analysis focuses on the nervous system. The *C. elegans* nerve ring is composed of 302 cells. How and when the nerve ring activates throughout late-stage development is largely unknown. The seam cells again act as fiducial markers to describe posture. Rapid imaging (3 Hz) enables observation of the embryo moving in the eggshell. Tracking the posture is defined as tracking all seam cells together. Patterns in movement (i.e. patterns in posture transformations) throughout late-stage development are used to understand nervous system development. Aforementioned twitching motions challenge existing object tracking methods. The high capture rate also necessitates a lose light dose to avoid photobleaching, making nuclei appear dim and close together.

Image capture rate is the key distinction between imaging approaches. The five minutes between images imposes a complete independence of postures between images, whereas images captured at 3 Hz show continuous motion of the embryo. Posture as a

composition of identified seam cells is used in both projects to mitigate embryonic twitching and conduct analyses in late-stage development. Differing facets of each imaging approach lead to the development of novel algorithms for object identification and multiple object tracking. Each algorithm is then applied to posture identification and posture tracking, respectively.

We explored the relationship between models for point-set identification and accompanying algorithms for *solving* the underlying optimization problem. Algorithms such as the Hungarian method [39] and Jonker-Volgenant [35] solve the LAP (Chapter 2.1.1) in polynomial time, enabling efficient application of simple models to large point-set matching tasks. Progressing from linear to quadratic models incurs a significant computational burden; exactly solving the QAP (Chapter 2.1.2) is $\mathcal{NP}$-hard. Heuristic approximation algorithms are polynomial, but offer no guarantee on the gap between a converged and optimal solution (Chapter 2.1.2.1). Assignment problems of higher orders (cubic, quartic, etc.) further suffer from burdensome computation. Some heuristic methods for the QAP have been adapted to solve higher order assignment problems, but even such methods are not readily applicable to assignment problems higher than a 4^{th} order model [27, 44, 95].

The lack of an existing algorithm which could flexibly integrate a wide range of matching models inspired EHGM (Chapter 3). Such an algorithm was necessary for pursuing posture identification, as biologically relevant cues used by biologists comprised the entire embryo. EHGM adapts a branch-and-bound approach to exactly solve arbitrarily intricate assignment problems [42]. The method was designed as an independent tool for approaching challenging point-set matching tasks. EHGM allows for models to be both rigorously tested due to the exactness of the method (Chapter A.1) and easily altered on

account of the flexibility of the branching optimization process. However, EHGM's worst case computational complexity is exponential in the number of points. The method is a novel contribution to assignment problem methodology and stands to improve performance on moderately sized point-set matching tasks.

We then applied EHGM to posture identification using hypergraphical models with features derived from experts performing manual annotation (Chapter 3.2.1). The hypergraphical model parameters were estimated using over 1200 instances of manually identified posture (Chapter 3.5.3). The *Posture* model solved with EHGM identified posture more than twice as often as an existing graphical method (Chapter 3.2.2) [96].

Posture identification enables subsequent cell tracking in late-stage development (Chapter 2). Detected cell nuclei are remapped to a "straightened" posture space which mitigates embryo-wise displacement of cell positions, i.e. movement that is due to twitching. The remapped coordinates can then be tracked using the GNN. The two-chapter sequence, Chs. 3 and 2, describes the application of state-of-the-art image segmentation and assignment problem methodologies to improve efficiency in completing the 4D atlas project [21].

The five minute interval between images contributes largely to the challenge in posture identification; the step size in conjunction with twitching ensures that posture at each timepoint is independent of preceding postures. While this non-traditional imaging approach ensures embryo health, it precludes application of standard point-set matching algorithms. A more traditional imaging style was applied to observe behavioral progression in *C. elegans* embryos (Chapter 2). The sensitive embryo was imaged at 3 Hz over the last 5 hours of development; the approach necessitated a low dose to preserve embryo-health. Each embryo imaged in this manner yields ≈54000 image volumes. The seam cell nuclei appear as dim

fluorescent spheroids moving in close proximity as the worm coils (Figure 2.8). Images from previous frames give context to seam cell nuclei identities at the present frame despite random bouts of twitching.

Posture tracking was approached as a MOT problem; however, existing MOT methods did not allow the modeling of interdependencies between object trajectories. Many MOT tasks can be successfully modeled via linear models in the same sense that traditional point-set matching problems can be adequately solved via linear and quadratic models. Moreover, objects do not reliably move together in many real-life MOT tasks; many image capture paradigms also enable observation of smooth predictable motion. Again, seam cell identification emerges as a cutting edge problem in the field.

MHT stands as the gold-standard paradigm for approaching an MOT task. Intersecting trajectories are disambiguated using future frame data under the assumption of smooth motion. Treating object movements as independent can yield strong results even if objects move interdependently. We were interested in intertwining MHT's deferred decision logic with our empirically validated hypergraphical modeling approach to yield a new tool for MOT tasks featuring correlated movement, such as posture tracking.

MHHT efficiently integrates MHT's multiple frame evaluation protocol with flexible hypergraphical modeling to track a hypergraphical form across several future frames. While recent MHT applications in fluorescence microscopy solve a large multidimensional assignment problem [17, 30], traditional implementations use an optimized Murty's algorithm [54, 55] to recursively update hypotheses across a specified number of future frames. An explicit tree search will consider at most K^N hypotheses, whereas using an optimized Murty method enables evaluating only KN hypotheses [23, 54]. The tree search method has the potential

to find a path with lower total hypergraphical cost of the K original hypotheses than the approximation at expense of computation; the tree search method is worst-case exponential in N while the approximation is worst-case (and average-case) linear in N (Chapter 2.5.1.3). MHHT's hypergraphical model evaluation step occurs *after* hypothesis generation via a linear model. As a result, the algorithm's scaling is largely independent of the chosen hypergraphical model.

MHHT with data-driven hypergraphical models improved posture tracking accuracy over baseline methods across levels of detection (Chapter 2.2.2). MHHT was relatively more effective on data with perfect detections, particularly on frames in the top quartile of frame-to-frame embryo displacement (Table 2.2). Volumetric imaging at 3 Hz stood to damage both the embryo and fluorophores; a low laser dose was used to compensate. However, the trade off caused nuclei to appear dim and close together, yielding a challenging detection problem. Nuclei were often under-segmented (undetected) or over-segmented (clumped together) despite the training and deployment of a 3D CNN (Chapter 2.5.2.2). MHHT dramatically lowered the error rate on frames with less movement, but also struggled on high movement frames (Figure 2.4). Nonetheless, MHHT proved to be more effective over baseline method at all levels of embryo-movement. We also developed a Python interface for posture tracking to enable efficient posture annotation of future imaged embryos. The algorithm and interface are easily applicable to other interdependent MOT tasks. Subsequent analysis of behavioral maturation gave insight into how and when the nervous system assumes control of locomotion [2].

Altogether, presented contributions highlight the application of image processing techniques and novel combinatorial optimization approaches. Methods can be broadly

described as approaching point-set identification tasks through the lens of supervised learning. Novel algorithms EHGM and MHHT integrate hypergraphical modeling in a data-driven fashion to point-set matching tasks. Developed methods were applied to identify a set of skin cells in embryonic *C. elegans*, a model organism for neurodevelopment. We demonstrated the efficacy of both EHGM and MHHT by dramatically improving posture identification accuracy and enabling more efficient analysis of posture data, respectively. The dynamism of both methods allows for applicability to domains and problems beyond seam cell identification in embryonic *C. elegans*.

This thesis bridged academic communities at two levels. First, the supervised learning and combinatorial optimization communities within the larger mathematics community were brought together in modeling cutting-edge point-set matching tasks. Second, the mathematical communities has been bridged with the developmental biology and microscopy communities. This partnership highlighted the capacity of multidisciplinary research in discovering biological phenomena.

Appendix A: Appendix

A.1 Exact Hypergraph Matching for Posture Identification in Embryonic *C. elegans*

EHGM Pseudocode

EHGM requires as input the branching step $k \in \{1, 2, \ldots, n_1\}$, the dissimilarity tensors $\mathbf{Z}^{(d)}$, $d = 1, 2, \ldots, n_1$, the size k permutation set $\mathbf{P}$, and optionally an initial upper bound C_0 on the global minimum C^*. The dissimilarity tensors are calculated given the reference hypergraph of size n_1, and either a previous frame hypergraph or a template hypergraph as described in *Model Fitting*. The lower degree dissimilarity tensors $\mathbf{Z}^{(d)}, d \leq 2k$ are calculated prior to the search and used to select branches. The higher degree dissimilarity terms $d > 2k$ are calculated during the search as required. Algorithm 1 initializes the search from the first candidate set $\mathbf{Q}_1 = \mathbf{P}$. The search is parallelized via initializing several first branches. Each explores a disjoint section of the domain $\mathcal{X}$.

Algorithm 1 initializes arrays and variables to start the recursive branch search (Algorithm 2). Eligible branch candidates are subset from the general queue $\mathbf{P}$ into $\mathbf{Q}_m$ via the *Enqueue* procedure (Algorithm 3). Each $\mathbf{Q}_m$ contains the potential assignments for the next k terms that satisfy both the pruning constraints and assignment constraints

specified by $\mathcal{X}$. The current assignment cost $\tilde{C}$ is checked against the current minimum C^* upon reaching a complete assignment. The Backtrack procedure (Algorithm 4) removes $\mathbf{k}_{m-1}$ from $\mathbf{Q}_{m-1}$ when the path from $\mathbf{k}_{m-1}$ is exhausted, which occurs when $\mathbf{Q}_m = \emptyset$. The recursion will continue until $\mathbf{Q}_1$ is empty, signaling the complete enumeration of the search space S_n.

Algorithm 1: EHGM

Input: $k, C_0, \mathbf{P}, \mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(2k)}$

Output: $\mathbf{x}^*, C^* = f(\mathbf{x}^*)$

Initialization

$C^* \leftarrow C_0$

$\tilde{\mathbf{H}} = []$

$\tilde{\mathbf{I}} = []$

$\tilde{\mathbf{x}} \leftarrow \emptyset$

$\tilde{C} \leftarrow 0$

$m \leftarrow 1$

$\mathbf{Q}_1 \leftarrow Enqueue(\tilde{\mathbf{x}}, \mathbf{P}, \tilde{C}, C^*, 1)$

while $\mathbf{Q}_1 \neq \emptyset$ **do**

$\mathbf{k}_1 \leftarrow \mathbf{Q}_1.pop()$

$\hat{H}_1 \leftarrow H_1(\mathbf{k}_1 | \mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(k)})$

$\tilde{C} \leftarrow \hat{H}_1$

$\tilde{\mathbf{H}}[1] = \hat{H}_1$

$\tilde{\mathbf{x}} \leftarrow \tilde{\mathbf{x}} \cup \{\mathbf{k}_1\}$

$m \mathrel{+}= 1$

$Visit(\mathbf{P}, \tilde{\mathbf{x}}, \tilde{C}, C^*, m)$

end

Return: $\mathbf{x}^*, C^*$

Algorithm 2: Visit

Input: $\mathbf{P}, \tilde{\mathbf{x}}, \tilde{C}, C^*, m$

$\mathbf{Q}_m \leftarrow Enqueue(\tilde{\mathbf{x}}, \mathbf{P}, \tilde{C}, C^*, m)$

while $\mathbf{Q}_m \neq \emptyset$ **do**

$\hat{\mathbf{k}}_m \leftarrow \mathbf{Q}_m.pop()$

$\tilde{\mathbf{x}} \leftarrow \tilde{\mathbf{x}} \cup \{\hat{\mathbf{k}}_m\}$

$m \mathrel{+}= 1$

$\hat{H}_m \leftarrow H_m(\mathbf{k}_m|\tilde{\mathbf{x}}, \mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(2k)})$

$\tilde{\mathbf{H}}[m] = \hat{H}_m$

$\tilde{C} \mathrel{+}= \hat{H}_m$

if $m \geq 3$ **then**

$\hat{I}_m \leftarrow I_m(\mathbf{k}_m|\tilde{\mathbf{x}}, \mathbf{Z}^{(2k+1)}, \ldots, \mathbf{Z}^{(mk)})$

$\tilde{\mathbf{I}}[m] = \hat{I}_m$

$\tilde{C} \mathrel{+}= \hat{I}_m$

end

if $m < M$ **then**

$Visit(\mathbf{P}, \tilde{\mathbf{x}}, \tilde{C}, C^*, m)$

else if $m = M$ **then**

if $\tilde{C} \leq C^*$ **then**

$\mathbf{x}^* \leftarrow \tilde{\mathbf{x}}$

$C^* \leftarrow \tilde{C}$

end

$Backtrack(\tilde{\mathbf{x}}, \mathbf{Q}_{m-1}, m)$

end

Algorithm 3: Enqueue

Input: $\tilde{\mathbf{x}}, \mathbf{P}, \tilde{C}, C^*, m$

Output: $\mathbf{Q}_m$

$\mathbf{Q}_m \leftarrow \emptyset$

for $\mathbf{k} \in \mathbf{P}$ **do**

  if $(\mathbf{k} \cap \tilde{\mathbf{x}} = \emptyset) \wedge (\tilde{C} + H_m(\mathbf{k}|\tilde{\mathbf{x}}) < C^*)$ **then**

    $\mathbf{Q}_m \leftarrow \mathbf{Q}_m \cup \mathbf{k}$

  end

end

Algorithm 4: Backtrack

Input: $\tilde{\mathbf{x}}, \mathbf{Q}_{m-1}, m$

$\mathbf{Q}_{m-1} \leftarrow \mathbf{Q}_{m-1} \setminus \tilde{\mathbf{x}}_m$

$\tilde{\mathbf{x}} \leftarrow \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_{m-1}\}$

$\tilde{C} \leftarrow \Sigma_{j=1}^{m-1}(\mathbf{C}_m + \mathbf{I}_m)$

Hypergraphical Objective Decomposition

The hypergraphical optimization objective can be decomposed according to hyperedge multiplicity and branching step. The stratification enables efficient search via EHGM.

Theorem 1. *Assume an assignment problem objective f is in the form:*

$$f(X|\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},\ldots,\mathbf{Z}^{(n_1)}) = \sum_{l_1=1}^{n_1}\sum_{l_1'=1}^{n_2}\mathbf{Z}^{(1)}_{l_1l_1'}x_{l_1l_1'} + \sum_{l_1=1}^{n_1}\sum_{l_1'=1}^{n_2}\sum_{l_2=l_1+1}^{n_1}\sum_{l_2'=1}^{n_2}\mathbf{Z}^{(2)}_{l_1l_1'l_2l_2'}x_{l_1l_1'}x_{l_2l_2'}$$
$$+\sum_{l_1=1}^{n_1}\sum_{l_1'=1}^{n_2}\sum_{l_2=l_1+1}^{n_1}\sum_{l_2'=1}^{n_2}\sum_{l_3=l_2+1}^{n_1}\sum_{l_3'=1}^{n_2}\mathbf{Z}^{(3)}_{l_1l_1'l_2l_2'l_3l_3'}x_{l_1l_1'}x_{l_2l_2'}x_{l_3l_3'} + \ldots$$
$$+\sum_{l_1=1}^{n_1}\sum_{l_1'=1}^{n_2}\ldots\sum_{l_{n_1}=l_{n_1-1}+1}^{n_1}\sum_{l_{n_1}'=1}^{n_2}\mathbf{Z}^{(n_1)}_{l_1l_1'\ldots l_{n_1}l_{n_1}'}x_{l_1l_1'}\ldots x_{l_{n_1}l_{n_1}'} \quad \text{(A.1)}$$

Then, for $k \in \{1,2,\ldots,n_1\}$*, the stratification fully describes the objective* f *after* $M = \frac{n_1}{k}$ *branches. Define of* H_1*,* H_m*, and* I_m*:*

$$H_1(\mathbf{K}_1|\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},...,\mathbf{Z}^{(k)}) :=$$
$$\sum_{i_1=1}^{k}\mathbf{Z}^{(1)}_{l_{i_1}l_{i_1}'} + \sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}\mathbf{Z}^{(2)}_{l_{i_1}l_{i_1}'l_{i_2}l_{i_2}'} + ... + \sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}\cdots\sum_{i_k=i_{k-1}+1}^{k}\mathbf{Z}^{(k)}_{l_{i_1}l_{i_1}'l_{i_2}l_{i_2}'\ldots l_{i_k}l_{i_k}'}$$

$$H_m(\mathbf{K}_m|\mathbf{K}_1,...,\mathbf{K}_{m-1},\mathbf{Z}^{(1)},...,\mathbf{Z}^{(2k)}) :=$$
$$\sum_{i_1=(m-1)k+1}^{mk}\mathbf{Z}^{(1)}_{l_{i_1}l_{i_1}'} + \sum_{i_2=(m-1)k+1}^{mk}\sum_{i_1=1}^{i_2-1}\mathbf{Z}^{(2)}_{l_{i_1}l_{i_1}'l_{i_2}l_{i_2}'}$$
$$+\sum_{i_3=(m-1)k+1}^{mk}\sum_{i_2=1}^{i_3-1}\sum_{i_1=1}^{i_2-1}\mathbf{Z}^{(3)}_{l_{i_1}l_{i_1}'l_{i_2}l_{i_2}'l_{i_3}l_{i_3}'} + ... + \sum_{i_{2k}=(m-1)k+1}^{mk}\sum_{i_{2k-1}=1}^{i_{2k}}\cdots\sum_{i_1=1}^{i_2-1}\mathbf{Z}^{(2k)}_{l_{i_1}l_{i_1}'\ldots l_{i_{2k}}l_{i_{2k}}'}$$

$$I_m(\mathbf{K}_m|\mathbf{K}_1,\mathbf{K}_2,\ldots,\mathbf{K}_{m-1},\mathbf{Z}^{(2k+1)},\ldots,\mathbf{Z}^{(mk)}) := \sum_{d=2k+1}^{mk}\Xi_m^{(d)}$$

where

$$\Xi_m^{(d)}(\mathbf{K}_m|\mathbf{K}_1,\mathbf{K}_2,\ldots,\mathbf{K}_{m-1},\mathbf{Z}^{(2k+1)},\ldots,\mathbf{Z}^{(mk)}) := \sum_{i_d=(m-1)k+1}^{mk}\sum_{i_{d-1}=1}^{i_d-1}\cdots\sum_{i_1=1}^{i_2-1}\mathbf{Z}^{(d)}_{l_{i_1}l'_{i_1}\ldots l_{i_d}l'_{i_d}}$$

Then, the degree n_1 hypergraph matching objective f can be expressed

$$f(X|\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},\ldots,\mathbf{Z}^{(n_1)}) = \sum_{m=1}^{m} H_m + \sum_{m=3}^{m} I_m$$

Proof. First consider the single branching case $k = 1$. This yields $M = \frac{n_1}{k} = \frac{n_1}{1} = n_1$ branches. Each branch yields one assignment; i.e. $K_m = l'_m$ is assigned to the vertex l_m. The initial branch selection rule H_1 can only utilize the first order term:

$$H_1(K_1|\mathbf{Z}^{(1)}) = \mathbf{Z}^{(1)}_{l_1l'_1}$$

Then the general selection rule for the second branch will: gather the first order costs for the second assignment as well as the quadratic (second order) costs between the first two assignments:

$$H_2(K_2|K_1,\mathbf{Z}^{(1)},\mathbf{Z}^{(2)}) = \mathbf{Z}^{(1)}_{l_2l'_2} + \sum_{i_2=2}^{2}\sum_{i_1=1}^{i_2}\mathbf{Z}^{(2)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}} = \mathbf{Z}^{(1)}_{l_2} + \mathbf{Z}^{(2)}_{l_1l'_1l_2l'_2}$$

The third branching step will include H_3 and I_3. H_3 follows from H_2:

$$H_3(K_3|K_1,K_2,\mathbf{Z}^{(1)},\mathbf{Z}^{(2)}) = \mathbf{Z}^{(1)}_{l_3l'_3} + \sum_{i_2=3}^{3}\sum_{i_1=1}^{i_2-1}\mathbf{Z}^{(2)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}} = \mathbf{Z}^{(1)}_{l_3l'_3} + \mathbf{Z}^{(2)}_{l_1l'_1l_3l'_3} + \mathbf{Z}^{(2)}_{l_2l'_2l_3l'_3}$$

$$I_3(K_3|K_1,K_2,\mathbf{Z}^{(3)}) = \Xi^{(3)}_m = \mathbf{Z}^{(3)}_{l_1l'_1l_2l'_2l_3l'_3}$$

Note that if $n_1 = 3$, then $H_1 + H_2 + H_3 + I_3$ fully describes the third order assignment problem:

$$\begin{aligned}
&H_1 + H_2 + H_3 + I_3 \\
&\quad= \underbrace{\mathbf{Z}^{(1)}_{l_1l'_1}}_{H_1} + \underbrace{\mathbf{Z}^{(1)}_{l_2l'_2} + \mathbf{Z}^{(2)}_{l_1l'_1l_2l'_2}}_{H_2} + \underbrace{\mathbf{Z}^{(1)}_{l_3l'_3} + \mathbf{Z}^{(2)}_{l_1l'_1l_3l'_3} + \mathbf{Z}^{(2)}_{l_2l'_2l_3l'_3}}_{H_3} + \underbrace{\mathbf{Z}^{(3)}_{l_1l'_1l_2l'_2l_3l'_3}}_{I_3} \\
&\quad= \sum_{i_1=1}^{3}\mathbf{Z}^{(1)}_{l_{i_1}l'_{i_1}} + \sum_{i_1=1}^{3}\sum_{i_2=i_1+1}^{3}\mathbf{Z}^{(2)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}} + \sum_{i_1=1}^{3}\sum_{i_2=i_1+1}^{3}\sum_{i_3=i_2+1}^{3}\mathbf{Z}^{(3)}_{l_{i_1}l'_{i_1}l_{i_2}l'_{i_2}l_{i_3}l'_{i_3}} \\
&\quad= f(X|\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},\mathbf{Z}^{(3)}) \quad \text{(A.2)}
\end{aligned}$$

Now consider the extension to $n_1 = 4$, yielding a fourth degree assignment problem. The fourth branch will assign the next term, $K_4 = l'_4$. The terms H_4 and I_4 will then fully specify the fourth degree problem:

$$H_4(K_4|K_1,K_2,K_3,\mathbf{Z}^{(1)},\mathbf{Z}^{(2)}) = \mathbf{Z}^{(1)}_{l_4l'_4} + \mathbf{Z}^{(2)}_{l_1l'_1l_4l'_4} + \mathbf{Z}^{(2)}_{l_2l'_2l_4l'_4} + \mathbf{Z}^{(2)}_{l_3l'_3l_4l'_4}$$

The second aggregation rule I_4 will consider third order terms between branches $1, 2$

and 4 as well as the fourth order term using all four assignments:

$$I_4(K_4|K_1,K_2,K_3,\mathbf{Z}^{(3)},\mathbf{Z}^{(4)}) = \Xi_4^{(3)} + \Xi_4^{(4)} = \mathbf{Z}^{(3)}_{l_1 l'_1 l_2 l'_2 l_4 l'_4} + \mathbf{Z}^{(3)}_{l_2 l'_2 l_3 l'_3 l_4 l'_4} + \mathbf{Z}^{(4)}_{l_1 l'_1 l_2 l'_2 l_3 l'_3 l_4 l'_4}$$

Joining the fourth branch:

$$\begin{aligned}
H_1 + H_2 + H_3 + I_3 + H_4 + I_4 = \\
&\underbrace{\sum_{i_1=1}^{3} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{3}\sum_{i_2=i_1+1}^{3} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \sum_{i_1=1}^{3}\sum_{i_2=i_1+1}^{3}\sum_{i_3=i_2+1}^{3} \mathbf{Z}^{(3)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2} l_{i_3} l'_{i_3}}}_{H_1+H_2+H_3+I_3} \\
&+ \underbrace{\mathbf{Z}^{(1)}_{l_4 l'_4} + \mathbf{Z}^{(2)}_{l_1 l'_1 l_4 l'_4} + \mathbf{Z}^{(2)}_{l_2 l'_2 l_4 l'_4} + \mathbf{Z}^{(2)}_{l_3 l'_3 l_4 l'_4}}_{H_4} + \underbrace{\mathbf{Z}^{(3)}_{l_1 l'_1 l_2 l'_2 l_4 l'_4} + \mathbf{Z}^{(3)}_{l_2 l'_2 l_3 l'_3 l_4 l'_4} + \mathbf{Z}^{(4)}_{l_1 l'_1 l_2 l'_2 l_3 l'_3 l_4 l'_4}}_{I_4} \\
&= \sum_{i_1=1}^{4} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{4}\sum_{i_2=i_1+1}^{4} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \sum_{i_1=1}^{4}\sum_{i_2=i_1+1}^{4}\sum_{i_3=i_2+1}^{4} \mathbf{Z}^{(3)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2} l_{i_3} l'_{i_3}} \\
&\quad + \sum_{i_1=1}^{4}\sum_{i_2=i_1+1}^{4}\sum_{i_3=i_2+1}^{4}\sum_{i_4=i_3+1}^{4} \mathbf{Z}^{(4)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2} l_{i_3} l'_{i_3} l_{i_4} l'_{i_4}} \\
&= f(X|\mathbf{Z}^{(1)},\mathbf{Z}^{(2)},\mathbf{Z}^{(3)},\mathbf{Z}^{(4)}) \quad \text{(A.3)}
\end{aligned}$$

Now consider the arbitrary $(m+1)^{st}$ branch. This will yield the full objective for an assignment problem of size $m+1$ up to degree $m+1$.

$$\sum_{p=1}^{m+1} H_p + \sum_{p=3}^{m+1} I_p = \sum_{p=1}^{m} H_p + \sum_{p=3}^{m} I_p + H_{m+1} + I_{m+1} =$$

$$\underbrace{\sum_{i_1=1}^{m} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{m} \sum_{i_2=i_1+1}^{m} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \cdots + \sum_{i_1=1}^{m} \sum_{i_2=i_1+1}^{m} \cdots \sum_{i_m=i_{m-1}+1}^{m} \mathbf{Z}^{(m)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2} \ldots l_{i_m} l'_{i_m}}}_{\sum_{p=1}^{m} H_p + \sum_{p=3}^{m} I_p}$$

$$+ \underbrace{\mathbf{Z}^{(1)}_{l_{i_{m+1}} l'_{i_{m+1}}} + \sum_{i_1=1}^{m} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{m+1} l'_{m+1}}}_{H_{m+1}} + \underbrace{\sum_{d=3}^{m+1} \Xi^{(d)}_{m+1}}_{I_{m+1}} \quad \text{(A.4)}$$

It is sufficient to show each degree $d \in \{1, 2, \ldots, m+1\}$ hyperedge is fully accounted for across all $m+1$ points to prove the $(m+1)^{st}$ branch satisfies the objective f. The hyperedge costs across all points will be decomposed into three disjoint sets, and each set considered at a time:

$$\{1\}, \{2\}, \{3, \ldots, m\}, \{m+1\}$$

The first and final of the four cases are trivial. The first degree terms are enumerated via the first term in H_{m+1}, while $Xi^{(m+1)}_{m+1}$ explicitly addresses the degree $m+1$ hyperedge comprising all assignments: $\mathbf{Z}^{(m+1)}_{l_1 l'_1 l_2 l'_2 \ldots l_{m+1} l'_{m+1}}$. We will focus on the second and third cases. The degree $d = 2$ terms are formed by the addition of branch $m+1$ are considered in term H_{m+1}:

$$\sum_{i_1=1}^{m} \sum_{i_2=i_1+1}^{m} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \sum_{i_1=1}^{m} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_{m+1}} l'_{i_{m+1}}} = \sum_{i_1=1}^{m+1} \sum_{i_2=i_1+1}^{m+1} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}}$$

Let $d \in \{3, \ldots, m\}$. The completion is similar to the $d = 2$ degree case; however,

the term $\Xi_{m+1}^{(d)}$ in I_{m+1} address higher degree hyperedges up to and including degree m concerning branch $m+1$:

$$\sum_{i_1=1}^{m} \dots \sum_{i_d=i_{d-1}+1}^{m} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \dots l_{i_d} l'_{i_d}} + \Xi_{m+1}^{(d)} = \sum_{i_1=1}^{m+1} \dots \sum_{i_d=i_{d-1}+1}^{m+1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \dots l_{i_d} l'_{i_d}}$$

Therefore, the $(m+1)^{st}$ step fully accrues the objective f:

$$\sum_{p=1}^{m+1} H_p + \sum_{p=3}^{m+1} I_p = f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \dots, \mathbf{Z}^{(m+1)})$$

Then inductively, the stratification holds such that:

$$\sum_{m=1}^{n_1} H_m + \sum_{m=3}^{n_1} I_m = f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \dots \mathbf{Z}^{(n_1)})$$

Now consider the plural branching rule $k > 1$. The proof will follow from the single assignment branching case. The base case at the fourth branch will be established, followed by the induction hypothesis demonstrating the branching from m to $m+1$. First, define the terms H_1, H_2, H_3, I_3, H_4, and I_4:

$$H_1 = \sum_{i_1=1}^{k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{k} \sum_{i_2=i_1+1}^{k} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \dots + \sum_{i_1=1}^{k} \sum_{i_2=i_1+1}^{k} \dots \sum_{i_k=i_{k-1}+1}^{k} \mathbf{Z}^{(k)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2} \dots l_{i_k} l'_{i_k}} \quad \text{(A.5)}$$

$$H_2 = \sum_{i_1=k+1}^{2k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_2=k+1}^{2k} \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \dots + \sum_{i_{2k}=k+1}^{2k} \dots \sum_{i_2=1}^{i_3-1} \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2k)}_{l_{i_1} l'_{i_1} \dots l_{i_{2k}} l'_{i_{2k}}} \quad \text{(A.6)}$$

$$H_3 = \sum_{i_1=2k+1}^{3k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_2=2k+1}^{3k} \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + ... + \sum_{i_{2k}=2k+1}^{3k} ... \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2k)}_{l_{i_1} l'_{i_1} ... l_{i_{2k}} l'_{i_{2k}}} \tag{A.7}$$

$$I_3 = \sum_{d=2k+1}^{3k} \Xi_3^{(d)} \tag{A.8}$$

$$H_4 = \sum_{i_1=3k+1}^{4k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_2=3k+1}^{4k} \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + ... + \sum_{i_{2k}=3k+1}^{4k} ... \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2k)}_{l_{i_1} l'_{i_1} ... l_{i_{2k}} l'_{i_{2k}}} \tag{A.9}$$

$$I_4 = \sum_{d=3k+1}^{4k} \Xi_4^{(d)} \tag{A.10}$$

The terms presented thus far for the general $k > 1$ case fully describe all terms concerning assignments $1, 2, \ldots 4k$ up to degree $4k$. The hyperedge multiplicities will again be partitioned into disjoint groups:

$$\{1\}, \{2, \ldots, k\}, \{k+1, \ldots, 2k\}, \{2k+1, \ldots, 3k\}, \{3k+1, \ldots, 4k\}$$

The first case is trivial, just as in the single assignment branching ($k = 1$) proof. Unary terms are accounted for in the first summand of each H_m. Then, consider $d \in \{2, \ldots, k\}$:

$$\underbrace{\sum_{i_1=1}^{k} \cdots \sum_{i_d=i_{d-1}+1}^{k} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{H_1} + \underbrace{\sum_{i_d=k+1}^{2k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{H_2}$$
$$+ \underbrace{\sum_{i_d=2k+1}^{3k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{H_3} + \underbrace{\sum_{i_d=3k+1}^{4k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{H_4}$$
$$= \sum_{i_1=1}^{4k} \sum_{i_2=i_1+1}^{4k} \cdots \sum_{i_d=i_{d-1}+1}^{4k} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}$$

The proof for degree $d \in \{k+1, \ldots, 2k\}$ follows immediately from the grouping presented above, but without the initial branch selection rule term H_1. Next, assume $d \in \{2k+1, \ldots, (m-1)k\}$. Degree d hyperedge dissimilarities will be contained in both I_3 and I_4 terms:

$$\underbrace{\sum_{i_d=2k+1}^{3k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{I_3} + \underbrace{\sum_{i_d=3k+1}^{4k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{I_4}$$
$$= \sum_{i_1=1}^{4k} \cdots \sum_{i_{d-1}=i_{d-2}+1}^{4k} \sum_{i_d=i_{d-1}+1}^{4k} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}$$

Since $d \leq 2k+1$, the terms only appear in the third branch term I_3 when the assignment $2k+1$ is committed. The final set arises from the definition of I_4 which accrues hyperedges of degree $d \in \{3k+1, \ldots, 4k\}$ across assignments in branches $m = 1, 2, 3, 4$. The base case is fully established for the arbitrary $k > 1$ case. The final step of the proof is to establish the extension of the $(m+1)^{st}$ branch:

$$\sum_{p=1}^{m+1} H_p + \sum_{p=3}^{m+1} I_p = \sum_{p=1}^{m} H_p + \sum_{p=3}^{m} I_p + H_{m+1} + I_{m+1} =$$

$$\underbrace{\sum_{i_1=1}^{mk} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \sum_{i_1=1}^{mk} \sum_{i_2=i_1+1}^{mk} \mathbf{Z}^{(2)}_{l_{i_1} l'_{i_1} l_{i_2} l'_{i_2}} + \cdots + \sum_{i_1=1}^{mk} \sum_{i_2=i_1+1}^{mk} \cdots \sum_{i_{mk}=i_{mk-1}+1}^{mk} \mathbf{Z}^{(mk)}_{l_{i_1} l'_{i_1} \ldots l_{i_{mk}} l'_{i_{mk}}}}_{\sum_{p=1}^{m} H_p + \sum_{p=3}^{m} I_p}$$

$$+ \underbrace{\sum_{i_1=mk+1}^{(m+1)k} \mathbf{Z}^{(1)}_{l_{i_1} l'_{i_1}} + \ldots + \sum_{i_{2k}=mk+1}^{(m+1)k} \sum_{i_{2k-1}=1}^{i_{2k}-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(2k)}_{l_{i_1} l'_{i_1} \ldots l_{i_{2k}} l'_{i_{2k}}}}_{H_{m+1}} + \underbrace{\sum_{d=2k+1}^{(m+1)k} \Xi^{(d)}_{m+1}}_{I_{m+1}} \quad \text{(A.11)}$$

The $(m+1)k$ hyperedge multiplicities will be stratified into four groups:

$$\{1\}, \{2, \ldots, 2k\}, \{2k+1, \ldots, mk\}, \{mk+1, \ldots, (m+1)k\}$$

Just as in the singular $k = 1$ case, the proof for the first and last groups are trivial. The unary terms are again evident from the first term in H_{m+1}, while the $mk+1 \leq d \leq (m+1)k$ terms in I_{m+1} fully encapsulates the fourth group. The steps in the remaining two cases will follow that of the $k = 1$ case.

First, assume $d \in \{2, \ldots, 2k\}$. The extension of the $(m+1)^{st}$ branch uses exclusively the selection rule H_{m+1}:

$$\underbrace{\sum_{i_1=1}^{mk} \cdots \sum_{i_{d-1}=i_{d-2}+1}^{mk} \sum_{i_d=i_{d-1}+1}^{mk} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{\sum_{p=1}^{m} H_m} + \underbrace{\sum_{i_d=mk+1}^{(m+1)k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{H_{m+1}} =$$

$$\sum_{i_1=1}^{(m+1)k} \cdots \sum_{i_{d-1}=i_{d-2}+1}^{(m+1)k} \sum_{i_d=i_{d-1}+1}^{(m+1)k} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}} \quad \text{(A.12)}$$

Next, assume $d \in \{2k+1, \ldots, mk\}$. These terms are captured in I_{m+1} using each definition of $\Xi^{(d)}_{m+1}$:

$$\sum_{i_1=1}^{mk} \cdots \sum_{i_{d-1}=i_{d-2}+1}^{mk} \sum_{i_d=i_{d-1}+1}^{mk} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}} + \underbrace{\sum_{i_d=mk+1}^{(m+1)k} \sum_{i_{d-1}=1}^{i_d-1} \cdots \sum_{i_1=1}^{i_2-1} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}}}_{I_{m+1}} =$$

$$\sum_{i_1=1}^{(m+1)k} \cdots \sum_{i_{d-1}=i_{d-2}+1}^{(m+1)k} \sum_{i_d=i_{d-1}+1}^{(m+1)k} \mathbf{Z}^{(d)}_{l_{i_1} l'_{i_1} \ldots l_{i_d} l'_{i_d}} \quad \text{(A.13)}$$

All four results together show that every degree hyperedge $1, \ldots, (m+1)k$ is accounted for in the $(m+1)^{st}$ branch, thus proving the induction hypothesis:

$$\sum_{p=1}^{m+1} H_p + \sum_{p=3}^{m+1} I_p = f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots \mathbf{Z}^{((m+1)k)})$$

The M^{th} branch completes the degree n_1 assignment problem of size n_1. For any $k \in \{1, 2, \ldots n_1\}$, the selection and aggregation rules yield the full degree n_1 assignment problem objective:

$$\sum_{p=1}^{M} H_p + \sum_{p=3}^{M} I_p = f(X|\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots \mathbf{Z}^{(n_1)})$$

□

Convergence & Exactness of EHGM

Theorem 2. *EHGM (algorithm 1) will converge to a globally optimal solution of the following hypergraph matching optimization problem given input $k \in \{1, 2, \ldots, n_1\}$:*

$$\min_{X \in X} \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \mathbf{Z}^{(1)}_{l_1 l'_1} x_{l_1 l'_1} + \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \sum_{l_2=l_1+1}^{n_1} \sum_{l'_2=1}^{n_2} \mathbf{Z}^{(2)}_{l_1 l'_1 l_2 l'_2} x_{l_1 l'_1} x_{l_2 l'_2}$$
$$+ \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \sum_{l_2=l_1+1}^{n_1} \sum_{l'_2=1}^{n_2} \sum_{l_3=l_2+1}^{n_1} \sum_{l'_3=1}^{n_2} \mathbf{Z}^{(3)}_{l_1 l'_1 l_2 l'_2 l_3 l'_3} x_{l_1 l'_1} x_{l_2 l'_2} x_{l_3 l'_3} + \ldots$$
$$+ \sum_{l_1=1}^{n_1} \sum_{l'_1=1}^{n_2} \ldots \sum_{l_{n_1}=l_{n_1-1}+1}^{n_1} \sum_{l'_{n_1}=1}^{n_2} \mathbf{Z}^{(n_1)}_{l_1 l'_1 \ldots l_{n_1} l'_{n_1}} x_{l_1 l'_1} \ldots x_{l_{n_1} l'_{n_1}} \quad \text{(A.14)}$$

where $\mathcal{X}$ is defined:

$$\mathcal{X} = \{X \in \{0,1\}^{n_1 \times n_2} : \forall j, \sum_{i=1}^{n_1} x_{ij} \leq 1, \forall i \sum_{j=1}^{n_2} x_{ij} = 1\} \quad \text{(A.15)}$$

Proof. First, we will show EHGM converges, then it will be proven that the converged solution is globally optimal.

The search begins with initializing queue $\mathbf{Q}_1 = \mathbf{P}$. The algorithm terminates with the exhaustion of $\mathbf{Q}_1$. Each set $\mathbf{Q}_m \subset \mathbf{P}$ contains feasible k-assignments conditioned on the assignment constraints and costs $\tilde{C}, C^*$. *Backtrack* (Algorithm 4) removes $\mathbf{x}_m$ from $\mathbf{Q}_{m-1}$ upon enumeration of $\mathbf{Q}_m$. The recursion then falls back to selecting from branch $m-1$, eventually exhausting $\mathbf{Q}_{m-1}$ just as in the enumeration of $\mathbf{Q}_m$. This recursion continues until the first branch $\mathbf{k}_1 \in \mathbf{Q}_1$ is removed, signaling the exploration of all assignments originating with the k-tuple $\mathbf{k}_1$. The exploration is repeated for each $\mathbf{k}_1 \in \mathbf{Q}_1$. Thus, all possible assignments $X \in \mathcal{X}$ are explored via the branching scheme.

Assignments accrue a monotonically increasing cost $\tilde{C}$ to be compared to C^* with accompanying assignment $\mathbf{x}^*$ at each branch. A complete assignment then drops the last k

assignments from $\tilde{\mathbf{x}}$, initializing the backwards recursion, emptying $\mathbf{Q}_m$ until $\mathbf{k}_{m-1}^{(n_k)} \in \mathbf{Q}_{m-1}$ is exhausted. There are at most $|\mathbf{Q}_m| \leq n_k$ viable permutations at branch m. Each possible branch is evaluated from $\tilde{\mathbf{x}} = [\mathbf{k}_1, \mathbf{k}_2, \ldots \mathbf{k}_{m-1}]$. The $(m-1)^{st}$ branch $\mathbf{k}_{m-1} \in \mathbf{Q}_{m-1}$ is removed from $\mathbf{Q}_{m-1}$ upon exhaustion of $\mathbf{Q}_m$:

$$\tilde{\mathbf{x}}_1^{(m)} = [\mathbf{k}_1^{(1)}, \mathbf{k}_2^{(1)}, \ldots \mathbf{k}_{m-1}^{(1)}, \mathbf{k}_m^{(1)}]$$

$$\tilde{\mathbf{x}}_2^{(m)} = [\mathbf{k}_1^{(1)}, \mathbf{k}_2^{(1)}, \ldots \mathbf{k}_{m-1}^{(1)}, \mathbf{k}_m^{(2)}]$$

$$\ldots$$

$$\tilde{\mathbf{x}}_{n_k}^{(m)} = [\mathbf{k}_1^{(1)}, \mathbf{k}_2^{(1)}, \ldots \mathbf{k}_{m-1}^{(1)}, \mathbf{k}_m^{(n_k)}]$$

Each of the n_k possible final branches from $\mathbf{k}_{m-1}^{(1)}$ is explored, then $\mathbf{k}_{m-1}^{(1)}$ is removed from $\mathbf{Q}_{m-1}$.

The process follows for the M^{th} branch, exhausting viable assignment sets until $\mathbf{k}_{M-1}$ is removed. The recursion follows inductively back to the exhaustion of $\mathbf{Q}_1$, signaling the end of the search. Thus, all possible assignments $X \in \mathcal{X}$ are explored via the branching scheme.

The convergent and exhaustive algorithm will yield a globally optimal solution $C^* = f(\mathbf{x}^*)$ after exhausting $\mathbf{Q}_1$. As proven above the additive decomposition of the cost structure (equation 3.2) is proven to be satisfied by summing all selection and aggregation rule terms. Assume an uninformed initialization $C^* = \infty$. Then the first pass will greedily take the

best permutation from the first set $\mathbf{Q}_1$: $\mathbf{k}_1^{(1)}$, and the best from the second set given it does not conflict with $\mathbf{k}_1^{(1)}$: $\mathbf{k}_2^{(2|1)}$. This process will continue until the first complete assignment: $\tilde{\mathbf{x}} = [\mathbf{k}_1^{(1)}, \mathbf{k}_2^{(2|1)}, \mathbf{k}_3^{(3|2,1)}, \ldots, \mathbf{k}_M^{(M|(M-1),\ldots,1)}]$ with $\tilde{C} = f(\tilde{\mathbf{x}})$. The first *Bracktrack* removes $\mathbf{k}_M^{(M|(M-1),\ldots,1)}$, and the M^{th} *Visit* call will exhaust $\mathbf{Q}_M$. Subsequent *Enqueue* calls will limit only allow branches that satisfy both the assignment constraints and the updated selection rule cost (Algorithm 3). This follows that any k-tuple of assignments $\mathbf{k}_m^{(j)}$ such that for $\tilde{\mathbf{x}} = [\mathbf{k}_1^{(j_1)}, \mathbf{k}_2^{(j_2|j_1)}, \ldots, \mathbf{k}_m^{(j_d|j_{(m-1)},\ldots j_1)}]$:

$$\tilde{C} + H_m(\tilde{\mathbf{x}}_{m-1}, \mathbf{k}_m^{(j_d|j_{(m-1)},\ldots j_1)}) < C^*$$

The additive decomposition of the objective paired with the assumed non-negativity of the dissimilarity tensors $\mathbf{Z}^{(j)}$ results in each branch monotonically increasing $\tilde{C}$:

$$\tilde{C} + H_m(\tilde{\mathbf{x}}_{m-1}, \mathbf{k}_m^{(j_d|j_{(m-1)},\ldots j_1)}) + I_m(\mathbf{k}_1^{(j_1)}, \mathbf{k}_2^{(j_2|j_1)}, \ldots, \mathbf{k}_m^{(j_d|j_{(m-1)},\ldots j_1)}) \geq \tilde{C}$$

The convergent search will thus eliminate all paths that are not globally optimal. Incrementally updating the reserved solution $\mathbf{x}^*$ with cost C^* expedites convergence as each replacement is necessarily a better solution. The resulting $\mathbf{x}^*$ and corresponding cost C^* are such that at no other full assignment $\tilde{\mathbf{x}}$ can replace $\mathbf{x}^*$, by definition a globally optimal solution of f.

□

Bibliography

[1] Evan L. Ardiel, Abhishek Kumar, Joseph Marbach, Ryan Christensen, Rishi Gupta, William Duncan, Jonathan S. Daniels, Nico Stuurman, Daniel Colón-Ramos, and Hari Shroff. Visualizing Calcium Flux in Freely Moving Nematode Embryos. *Biophysical Journal*, 112(9):1975–1983, May 2017.

[2] Evan L. Ardiel, Andrew Lauziere, Stephen Xu, Brandon J. Harvey, Ryan Christensen, Stephen Nurrish, Joshua M. Kaplan, and Hari Shroff. Systematic analysis of C. elegans embryo behavior reveals a stereotyped developmental progression and a rhythmic pattern of behavioral quiescence, December 2021. Pages: 2021.12.09.471955 Section: New Results.

[3] Zhirong Bao, John I. Murray, Thomas Boyle, Siew Loon Ooi, Matthew J. Sandel, and Robert H. Waterston. Automated cell lineage tracing in Caenorhabditis elegans. *Proceedings of the National Academy of Sciences of the United States of America*, 103(8):2707–2712, February 2006.

[4] Yaakov Bar-Shalom and Xiao-Rong Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. Yaakov Bar-Shalom, 3rd edition, 1995.

[5] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, December 1996.

[6] A.C. Berg, T.L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 26–33 vol. 1, June 2005. ISSN: 1063-6919.

[7] Dimitri P. Bertsekas. An Auction Algorithm for Shortest Paths. *SIAM Journal on Optimization*, July 2006. Publisher: Society for Industrial and Applied Mathematics.

[8] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468, September 2016. ISSN: 2381-8549.

[9] Garrett Birkhoff. Tres observaciones sobre el algebra lineal. *Univ. Nac. Tucumán*, Revista A:147–151, 1946.

[10] S. S. Blackman. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18, January 2004. Conference Name: IEEE Aerospace and Electronic Systems Magazine.

[11] Thomas J. Boyle, Zhirong Bao, John I. Murray, Carlos L. Araya, and Robert H. Waterston. AceTree: a tool for visual analysis of Caenorhabditis elegans embryogenesis. *BMC Bioinformatics*, 7(1):275, June 2006.

[12] Sydney Brenner. Nature's Gift to Science (Nobel Lecture). *ChemBioChem*, 4(8):683–687, 2003. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cbic.200300625.

[13] Rainer Burkard, Mauro Dell'Amico, and Silviano Martello. *Assignment Problems*. Society for Industrial Applied Mathematics, 2009.

[14] T.S. Caetano, J.J. McAuley, Li Cheng, Q.V. Le, and A.J. Smola. Learning Graph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1048–1058, June 2009.

[15] Jianfeng Cao, Guoye Guan, Vincy Wing Sze Ho, Ming-Kin Wong, Lu-Yan Chan, Chao Tang, Zhongying Zhao, and Hong Yan. Establishment of a morphological atlas of the Caenorhabditis elegans embryo using deep-learning-based 4D segmentation. *Nature Communications*, 11(1):6254, December 2020. Number: 1 Publisher: Nature Publishing Group.

[16] Martin Chalfie, Yuan Tu, Ghia Euskirchen, William W. Ward, and Douglas C. Prasher. Green Fluorescent Protein as a Marker for Gene Expression. *Science*, February 1994. Publisher: American Association for the Advancement of Science.

[17] N. Chenouard, I. Bloch, and J. Olivo-Marin. Multiple Hypothesis Tracking for Cluttered Biological Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2736–3750, November 2013. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[18] Michael Chertok and Yosi Keller. Efficient High Order Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2205–2215, December 2010.

[19] Minsu Cho, Jungmin Lee, and Kyoung Mu Lee. Reweighted Random Walks for Graph Matching. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, Lecture Notes in Computer Science, pages 492–505, Berlin, Heidelberg, 2010. Springer.

[20] Francois Chollet. Keras: Deep Learning for humans, September 2021. original-date: 2015-03-28T00:35:42Z.

[21] Ryan Patrick Christensen, Alexandra Bokinsky, Anthony Santella, Yicong Wu, Javier Marquina-Solis, Min Guo, Ismar Kovacevic, Abhishek Kumar, Peter W Winter, Nicole Tashakkori, Evan McCreedy, Huafeng Liu, Matthew McAuliffe, William Mohler, Daniel A Colón-Ramos, Zhirong Bao, and Hari Shroff. Untwisting the Caenorhabditis elegans embryo. *eLife*, 4:e10070, December 2015. Publisher: eLife Sciences Publications, Ltd.

[22] I. J. Cox and S. L. Hingorani. An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(2):138–150, February 1996. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[23] I.J. Cox and M.L. Miller. On finding ranked assignments with application to multitarget tracking and motion correspondence. *IEEE Transactions on Aerospace and Electronic Systems*, 31(1):486–489, January 1995.

[24] David F. Crouse. On implementing 2D rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, August 2016.

[25] Fabrice de Chaumont, Stéphane Dallongeville, Nicolas Chenouard, Nicolas Hervé, Sorin Pop, Thomas Provoost, Vannary Meas-Yedid, Praveen Pankajakshan, Timothée Lecomte, Yoann Le Montagner, Thibault Lagache, Alexandre Dufour, and Jean-Christophe Olivo-Marin. Icy: an open bioimage informatics platform for extended reproducible research. *Nature Methods*, 9(7):690–696, July 2012. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 7 Primary_atype: Reviews Publisher: Nature Publishing Group Subject_term: Bioinformatics;Imaging Subject_term_id: bioinformatics;imaging.

[26] OE Drummond, David Castañon, and MS Bellovin. Comparison of 2-D assignment algorithms for sparse, rectangular, floating point, cost matrices. In *SDI*, volume 4, pages 4–81, December 1990.

[27] Olivier Duchenne, Francis Bach, In-So Kweon, and Jean Ponce. A Tensor-Based Algorithm for High-Order Graph Matching. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, page 13, 2010.

[28] Emil Rudolph Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[29] A.X. Falcao, J. Stolfi, and R. de Alencar Lotufo. The image foresting transform: theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):19–29, January 2004. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[30] Linqing Feng, Yingke Xu, Yi Yang, and Xiaoxiang Zheng. Multiple dense particle tracking in fluorescence microscopy images based on multidimensional assignment. *Journal of Structural Biology*, 173(2):219–228, February 2011.

[31] T. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8(3):173–184, July 1983. Conference Name: IEEE Journal of Oceanic Engineering.

[32] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388, April 1996. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[33] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, October 2017. ISSN: 2380-7504.

[34] Khuloud Jaqaman, Dinah Loerke, Marcel Mettlen, Hirotaka Kuwata, Sergio Grinstein, Sandra L. Schmid, and Gaudenz Danuser. Robust single-particle tracking in live-cell time-lapse sequences. *Nature Methods*, 5(8):695–702, August 2008. Number: 8 Publisher: Nature Publishing Group.

[35] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340, December 1987.

[36] Richard M. Karp. Reducibility among Combinatorial Problems. In Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger, editors, *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department*, The IBM Research Symposia Series, pages 85–103. Springer US, Boston, MA, 1972.

[37] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, January 2017. arXiv: 1412.6980.

[38] Tjalling C. Koopmans and Martin Beckmann. Assignment Problems and the Location of Economic Activities. *Econometrica*, 25(1):53–76, 1957. Publisher: [Wiley, Econometric Society].

[39] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nav.3800020109.

[40] Abhishek Kumar, Yicong Wu, Ryan Christensen, Panagiotis Chandris, William Gandler, Evan McCreedy, Alexandra Bokinsky, Daniel A. Colón-Ramos, Zhirong Bao, Matthew McAuliffe, Gary Rondeau, and Hari Shroff. Dual-view plane illumination microscopy for rapid and spatially isotropic imaging. *Nature Protocols*, 9(11):2555–2573, November 2014.

[41] A. H. Land and A. G. Doig. An Automatic Method of Solving Discrete Programming Problems. *Econometrica*, 28(3):497–520, 1960. Publisher: [Wiley, Econometric Society].

[42] Andrew Lauziere, Ryan Christensen, Hari Shroff, and Radu Balan. An Exact Hypergraph Matching Algorithm for Nuclear Identification in Embryonic Caenorhabditis elegans, February 2022. arXiv:2104.10003 [cs, math].

[43] Eugene L. Lawler. The Quadratic Assignment Problem. *Management Science*, 9(4):586–599, 1963. Publisher: INFORMS.

[44] J. Lee, M. Cho, and K. M. Lee. Hyper-graph matching via reweighted random walks. In *CVPR 2011*, pages 1633–1640, June 2011. ISSN: 1063-6919.

[45] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 1482–1489 Vol. 2, Beijing, China, 2005. IEEE.

[46] Tony Lindeberg. Feature Detection with Automatic Scale Selection. *International Journal of Computer Vision*, 30(2):79–116, November 1998.

[47] John D. C. Little, Katta G. Murty, Dura W. Sweeney, and Caroline Karel. An Algorithm for the Traveling Salesman Problem. *Operations Research*, 11(6):972–989, December 1963. Publisher: INFORMS.

[48] Zhi-Yong Liu and Hong Qiao. GNCCP—Graduated NonConvexityand Concavity Procedure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1258–1267, June 2014. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[49] Benoit Lombardot. Interactive Watershed, May 2017. Section: Segmentation.

[50] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, June 2015. ISSN: 1063-6919.

[51] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[52] Daniel L. Mace, Peter Weisdepp, Louis Gevirtzman, Thomas Boyle, and Robert H. Waterston. A High-Fidelity Cell Lineage Tracing Method for Obtaining Systematic Spatiotemporal Gene Expression Patterns in Caenorhabditis elegans. *G3: Genes, Genomes, Genetics*, 3(5):851–863, May 2013. Publisher: G3: Genes, Genomes, Genetics Section: Investigations.

[53] M. J. McAuliffe, F. M. Lalonde, D. McGarry, W. Gandler, K. Csaky, and B. L. Trus. Medical Image Processing, Analysis and Visualization in clinical research. In *Proceedings 14th IEEE Symposium on Computer-Based Medical Systems. CBMS 2001*, pages 381–386, July 2001. ISSN: 1063-7125.

[54] M.L. Miller, H.S. Stone, and I.J. Cox. Optimizing Murty's ranked assignment method. *IEEE Transactions on Aerospace and Electronic Systems*, 33(3):851–862, July 1997.

[55] Katta G. Murty. An Algorithm for Ranking all the Assignments in Order of Increasing Cost. *Operations Research*, 16(3):682–687, 1968.

[56] Tomas Möller and Ben Trumbore. Fast, Minimum Storage Ray-Triangle Intersection. *Journal of Graphics Tools*, 2(1):21–28, January 1997. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10867651.1997.10487468.

[57] Quynh Nguyen, Francesco Tudisco, Antoine Gautier, and Matthias Hein. An Efficient Multilinear Optimization Framework for Hypergraph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1054–1075, June 2017. arXiv: 1511.02667.

[58] Jean-Christophe Olivo-Marin. Extraction of spots in biological images using multiscale products. *Pattern Recognition*, 35(9):1989–1996, September 2002.

[59] Dirk Padfield, Jens Rittscher, and Badrinath Roysam. Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis. *Medical Image Analysis*, 15(4):650–668, August 2011.

[60] William P. Pierskalla. Letter to the Editor—The Multidimensional Assignment Problem. *Operations Research*, 16(2):422–431, April 1968.

[61] J. R. Priess and D. I. Hirsh. Caenorhabditis elegans morphogenesis: the role of the cytoskeleton in elongation of the embryo. *Developmental Biology*, 117(1):156–173, September 1986.

[62] David M. Raizen, John E. Zimmerman, Matthew H. Maycock, Uyen D. Ta, Young-jai You, Meera V. Sundaram, and Allan I. Pack. Lethargus is a Caenorhabditis elegans sleep-like state. *Nature*, 451(7178):569–572, January 2008.

[63] Georgia Rapti. A perspective on C. elegans neurodevelopment: from early visionaries to a booming neuroscience research. *Journal of Neurogenetics*, 34(3-4):259–272, October 2020. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01677063.2020.1837799.

[64] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, December 1979. Conference Name: IEEE Transactions on Automatic Control.

[65] S. H. Rezatofighi, A. Milani, Z. Zhang, Q. Shi, A. Dick, and I. Reid. Joint Probabilistic Matching Using m-Best Solutions. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 136–145, June 2016. ISSN: 1063-6919.

[66] Seyed Hamid Rezatofighi, Anton Milan, Zhen Zhang, Qinfeng Shi, Anthony Dick, and Ian Reid. Joint Probabilistic Data Association Revisited. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3047–3055, Santiago, Chile, December 2015. IEEE.

[67] J. E. Richmond and E. M. Jorgensen. One GABA and two acetylcholine receptors function at the C. elegans neuromuscular junction. *Nature Neuroscience*, 2(9):791–797, September 1999.

[68] Donald L. Riddle, Thomas Blumenthal, Barbara J. Meyer, and James R. Priess. *C. elegans II*. Cold Spring Harbor Laboratory Press, 1997. Publication Title: C. elegans II. 2nd edition.

[69] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, Lecture Notes in Computer Science, pages 234–241, Cham, 2015. Springer International Publishing.

[70] Sartaj Sahni and Teofilo Gonzales. P-complete problems and approximate solutions. In *15th Annual Symposium on Switching and Automata Theory (swat 1974)*, pages 28–32, USA, October 1974. IEEE.

[71] Anthony Santella, Raúl Catena, Ismar Kovacevic, Pavak Shah, Zidong Yu, Javier Marquina-Solis, Abhishek Kumar, Yicong Wu, James Schaff, Daniel Colón-Ramos, Hari Shroff, William A. Mohler, and Zhirong Bao. WormGUIDES: an interactive single cell developmental atlas and tool for collaborative multidimensional data exploration. *BMC Bioinformatics*, 16(1):189, June 2015.

[72] Anthony Santella, Zhuo Du, and Zhirong Bao. A semi-local neighborhood-based framework for probabilistic cell lineage tracing. *BMC Bioinformatics*, 15(1):217, June 2014.

[73] Tim Scherr, Andreas Bartschat, Markus Resichl, Johannes Stegmaier, and Ralf Mikut. Best Practices in Deep Learning-Based Segmentation of Microscopy Images. In *Proceedings 28*, pages 175–195, Dortmund, Germany, November 2018.

[74] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, Jean-Yves Tinevez, Daniel James White, Volker Hartenstein, Kevin Eliceiri, Pavel Tomancak, and Albert Cardona. Fiji: an open-source platform for

biological-image analysis. *Nature Methods*, 9(7):676–682, July 2012. Number: 7 Publisher: Nature Publishing Group.

[75] Roman Spilger, Thomas Wollmann, Yu Qiang, Andrea Imle, Ji Young Lee, Barbara Müller, Oliver T. Fackler, Ralf Bartenschlager, and Karl Rohr. Deep Particle Tracker: Automatic Tracking of Particles in Fluorescence Microscopy Images Using Deep Learning. In Danail Stoyanov, Zeike Taylor, Gustavo Carneiro, Tanveer Syeda-Mahmood, Anne Martel, Lena Maier-Hein, João Manuel R.S. Tavares, Andrew Bradley, João Paulo Papa, Vasileios Belagiannis, Jacinto C. Nascimento, Zhi Lu, Sailesh Conjeti, Mehdi Moradi, Hayit Greenspan, and Anant Madabhushi, editors, *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Lecture Notes in Computer Science, pages 128–136, Cham, 2018. Springer International Publishing.

[76] Greg J. Stephens, Bethany Johnson-Kerner, William Bialek, and William S. Ryu. Dimensionality and dynamics in the behavior of C. elegans. *PLoS computational biology*, 4(4):e1000028, April 2008.

[77] J. E. Sulston, E. Schierenberg, J. G. White, and J. N. Thomson. The embryonic cell lineage of the nematode Caenorhabditis elegans. *Developmental Biology*, 100(1):64–119, November 1983.

[78] Shin-ya Takemura, Yoshinori Aso, Toshihide Hige, Allan Wong, Zhiyuan Lu, C Shan Xu, Patricia K Rivlin, Harald Hess, Ting Zhao, Toufiq Parag, Stuart Berg, Gary Huang, William Katz, Donald J Olbris, Stephen Plaza, Lowell Umayam, Roxanne Aniceto, Lei-Ann Chang, Shirley Lauchie, Omotara Ogundeyi, Christopher Ordish, Aya Shinomiya, Christopher Sigmund, Satoko Takemura, Julie Tran, Glenn C Turner, Gerald M Rubin, and Louis K Scheffer. A connectome of a learning and memory center in the adult Drosophila brain. *eLife*, 6:e26975, July 2017. Publisher: eLife Sciences Publications, Ltd.

[79] Jean-Yves Tinevez, Nick Perry, Johannes Schindelin, Genevieve M. Hoopes, Gregory D. Reynolds, Emmanuel Laplantine, Sebastian Y. Bednarek, Spencer L. Shorte, and Kevin W. Eliceiri. TrackMate: An open and extensible platform for single-particle tracking. *Methods*, 115:80–90, February 2017.

[80] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large Margin Methods for Structured and Interdependent Output Variables. *Journal of Machine Learning Research*, 6(50):1453–1484, 2005.

[81] P. Valdivia, P. Buono, C. Plaisant, N. Dufournaud, and J.-D. Fekete. Analyzing Dynamic Hypergraphs with Parallel Aggregated Ordered Hypergraph Visualization.

IEEE Transactions on Visualization and Computer Graphics, 27(1):1–13, January 2021. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[82] Abdulla Waleed. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow, 2017.

[83] Shaohe Wang, Stacy D. Ochoa, Renat N. Khaliullin, Adina Gerson-Gurwitz, Jeffrey M. Hendel, Zhiling Zhao, Ronald Biggs, Andrew D. Chisholm, Arshad Desai, Karen Oegema, and Rebecca A. Green. A high-content imaging approach to profile C. elegans embryonic development. *Development*, 146(7), April 2019. Publisher: Oxford University Press for The Company of Biologists Limited Section: TECHNIQUES AND RESOURCES.

[84] Tao Wang, Haibin Ling, Congyan Lang, and Songhe Feng. Graph Matching with Adaptive and Branching Path Following. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2853–2867, December 2018. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[85] Martin Weigart. StarDist - Object Detection with Star-convex Shapes, September 2021. original-date: 2018-06-26T19:54:45Z.

[86] Martin Weigert, Uwe Schmidt, Robert Haase, Ko Sugawara, and Gene Myers. Star-convex Polyhedra for 3D Object Detection and Segmentation in Microscopy. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3655–3662, Snowmass Village, CO, USA, March 2020. IEEE.

[87] L. Wen, W. Li, J. Yan, Z. Lei, D. Yi, and S. Z. Li. Multiple Target Tracking Based on Undirected Hierarchical Relation Hypergraph. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1282–1289, June 2014. ISSN: 1063-6919.

[88] J. G. White, D. G. Albertson, and M. A. Anness. Connectivity changes in a class of motoneurone during the development of a nematode. *Nature*, 271(5647):764–766, February 1978.

[89] John Graham White, Eileen Southgate, J. N. Thomson, and Sydney Brenner. The structure of the nervous system of the nematode Caenorhabditis elegans. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, 314(1165):1–340, November 1986. Publisher: Royal Society.

[90] Yicong Wu, Peter Wawrzusin, Justin Senseney, Robert S. Fischer, Ryan Christensen, Anthony Santella, Andrew G. York, Peter W. Winter, Clare M. Waterman, Zhirong

Bao, Daniel A. Colón-Ramos, Matthew McAuliffe, and Hari Shroff. Spatially isotropic four-dimensional imaging with dual-view plane illumination microscopy. *Nature Biotechnology*, 31(11):1032–1038, November 2013. Number: 11 Publisher: Nature Publishing Group.

[91] Junchi Yan, Changsheng Li, Yin Li, and Guitao Cao. Adaptive Discrete Hypergraph Matching. *IEEE Transactions on Cybernetics*, 48(2):765–779, February 2018. Conference Name: IEEE Transactions on Cybernetics.

[92] Yao Yao, Ihor Smal, Ilya Grigoriev, Anna Akhmanova, and Erik Meijering. Deep-learning method for data association in particle tracking. *Bioinformatics*, 36(19):4935–4941, December 2020.

[93] Tianshu Yu, Junchi Yan, Yilin Wang, and Wei Liu. Generalizing Graph Matching beyond Quadratic Assignment Model. In *NeurIPS*, page 11, 2018.

[94] Mikhail Zaslavskiy, Francis Bach, and Jean-Philippe Vert. A Path Following Algorithm for the Graph Matching Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242, December 2009. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[95] Ron Zass and Amnon Shashua. Probabilistic graph and hypergraph matching. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, AK, USA, June 2008. IEEE.

[96] Zhen Zhang, Yijian Xiang, Lingfei Wu, Bing Xue, and Arye Nehorai. KerGM: Kernelized Graph Matching. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, page 1. Curran Associates, Inc., 2019.

[97] F. Zhou and F. De la Torre. Factorized Graph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9):1774–1789, September 2016. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[98] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In Sebastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gozde Unal, and William Wells, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, Lecture Notes in Computer Science, pages 424–432, Cham, 2016. Springer International Publishing.